

Offline programování průmyslových robotů s využitím přístupu digitálního dvojčete

Offline Programming of Industrial Robots using the Digital Twin Approach

Bc. Filip Vicherek

Diplomová práce

Vedoucí práce: prof. Ing. Jiří Koziorek, Ph.D.

Ostrava, 2021

Abstrakt

Tato diplomová práce se zabývá offline programováním KUKA robotu s využitím SW Tecnomatix Process Simulate. V práci je provedena analýza trhu softwarů pro offline programování. Úloha je realizována na výukové buňce KUKA ready2 educate. Buňka obsahuje robotické rameno KUKA KR 3 R540, robotický kontrolér KR C4, aplikační prvky, PLC a HMI. Pro tento reálný model bylo vytvořeno digitální dvojče v programu Tecnomatix Process Simulate, ve kterém byla vytvořena kinematika ramene a chapadla. Vytvořené robotické operace využívají rozdílné nástroje a báze, typy pohybů a rychlosti. Výsledný program spouští dvě operace dle sepnutého signálu na HMI panelu. Výsledný program byl testován na reálném zařízení a digitálním dvojčeti, digitální model musel být upraven po testování, jelikož neodpovídal tomu reálnému.

Klíčová slova

diplomová práce, offline programování, robotické rameno, digitální dvojče, Tecnomatix Process Simulate, KUKA, KR 3 R540, KR C4

Abstract

This diploma thesis deals with offline programming of a KUKA robot using SW Tecnomatix Process Simulate. The thesis analyzes the market of software for offline programming. The task is performed on the KUKA ready2 educate training cell. The cell contains a robotic arm KUKA KR 3 R540, a robotic controller KR C4, application elements, PLC and HMI. A digital twin was created for this real model in program Tecnomatix Process Simulate, in which the kinematics of the arm and the gripper were created. Created robotic operations use different tools and bases, types of movements and speeds. The resulting program starts two operations according to the closed signal on the HMI panel. The resulting program was tested on a real device and a digital twin, the digital model had to be modified after testing, as it did not correspond to the real one.

Keywords

master thesis, offline programming, robotic arm, digital twin, Tecnomatix Process Simulate, KUKA, KR 3 R540, KR C4

Poděkování

Rád bych na tomto místě poděkoval vedoucímu práce prof. Ing. Jiří Koziorek, Ph.D. za odborné vedení, pomoc a rady při zpracování mé diplomové práce. Dále celé své rodině, která mě podporovala a poskytla mi prostor pro realizaci této práce.

Obsah

Seznam použitých symbolů a zkratk	7
Seznam obrázků	8
Seznam tabulek	10
1 Úvod	11
2 Offline programování robotů	13
2.1 KUKA.sim	13
2.2 Tecnomatix	14
2.3 OCTOPUZ	16
2.4 Visual Components	17
2.5 Delfoi	18
2.6 FASTSUITE	20
2.7 CIROS Studio	21
2.8 RobotDK	22
2.9 Robotmaster	23
2.10 AUTOMAPPPS	24
2.11 FAMOS	24
2.12 Verbotics Weld	25
2.13 SprutCAM Robot	26
3 Laboratorní pracoviště	28
3.1 KR 3 R540	28
3.2 Chapadlo	29
3.3 KUKA KR C4	31
3.4 KUKA smartPAD	31
3.5 Aplikační prvky	33
3.6 Uživatelské prostředí	34

3.7	PLC	36
3.8	HMI panel	37
3.9	Skupiny uživatelů	37
3.10	Volba pracovního režimu	38
4	Digitální dvojče robotické buňky	39
4.1	Import modelu	39
4.2	Robotické rameno KR 3 R540	40
4.3	Chapadlo	41
4.4	Montážní deska	41
4.5	Nastavení OLP v Process Simulate	42
4.6	Detekce kolizí	43
5	Návrh úlohy pro ověření offline programování	45
5.1	Návrh úlohy	45
5.2	Model světloometu	45
5.3	Konstrukce	46
6	Návrh a realizace programu	48
6.1	Návrh programu	48
6.2	Rozložení pracoviště	48
6.3	Tvorba operací	49
6.4	Path editor	50
6.5	Vytváření a úprava bodů	50
6.6	Parametrizace bodů	51
6.7	Typy pohybů	53
6.8	Robot configuration	55
6.9	OLP Configuration	57
6.10	Vytvořené trajektorie	57
6.11	Vytvoření programu robotu	60
6.12	Exportování programu	60
7	Ověření funkčnosti	62
7.1	Robot program viewer	62
7.2	Nahrání programu	62
7.3	Testování trajektorií na reálném zařízení	63
8	Zhodnocení výsledků	65
	Literatura	67

Přílohy	69
A Soubory	70

Seznam použitých zkratek a symbolů

OLP	– Offline programming
PS	– Process Simulate
CAD	– Computer aided design
CATIA	– Computer Aided Three Dimensional Interactive Application
JT	– Jupiter Tessellation
STEP	– Standard for the Exchange of Product Data
API	– Application Programming Interface
CAM	– Computer aided manufacturing
GUI	– Graphical User Interface
TCP	– Tool Central Point
WPS	– Welding Procedure Specification
STL	– Standard Tessellation Language
PLC	– Programmable Logic Controller
CNC	– Computer Numerical Control
HMI	– Human Machine Interface
OPC	– Ole for Process Control
OPC-UA	– OPC Unified Architecture
AGV	– Automated Guided Vehicle
CSV	– Comma-separated values
PLA	– Polylactic acid - Kyselina polymléčná
STL	– Stereolithography
PTP	– Point to point
LIN	– Linear
CIRC	– Circular
SPL	– Spline

Seznam obrázků

2.1	Prostředí KUKA.Sim [3]	14
2.2	Prostředí KUKA.OfficeLite [4]	15
2.3	Prostředí Tecnomatix Process Simulate	16
2.4	Prostředí OCTOPUZ [8]	17
2.5	Prostředí Visual Components [9]	18
2.6	Prostředí Delphi CUT [15]	19
2.7	Prostředí FastSuite [17]	21
2.8	Prostředí CIROS Studio [18]	22
2.9	Prostředí RobotDK [19]	23
2.10	Prostředí Robotmaster [20]	24
2.11	Prostředí AUTOMAPPPS [22]	25
2.12	Prostředí FAMOS [23]	26
2.13	Prostředí Verbotics Weld [24]	26
2.14	Prostředí SprutCAM Robot [25]	27
3.1	KUKA ready2_educate [27]	29
3.2	Ovládací a zobrazovací prvky buňky [27]	29
3.3	Pracovní zóny KR 3 R540 [28]	30
3.4	Chapadlo připevněné na robotu [27]	31
3.5	KR C4 Compact [27]	32
3.6	KUKA smartPAD	33
3.7	Základní aplikační prvky [27]	33
3.8	Vision aplikační prvky [27]	34
3.9	Vision systém [27]	35
3.10	Uživatelské prostředí [27]	36
3.11	Stavový řádek [27]	36
3.12	SIMATIC ET200 s CPU 1512SP-1	37
3.13	SIMATIC HMI KTP 900 Basic	37

4.1	Kinematika ramene v PS	40
4.2	Home pozice	40
4.3	Chapadlo a jeho kinematika v PS	41
4.4	Otevřená a zavřená pozice chapadla	42
4.5	Rozmístění bodů na montážní desce	42
4.6	Nastavení kontroléru	43
4.7	Rozmístění bodů na montážní desce	44
5.1	Díly podstav	46
5.2	Díly úchytů	46
5.3	Sestava světlometu	47
6.1	Rozložení pracoviště	49
6.2	Operation tree	50
6.3	Přidání KUKA-Krc	50
6.4	Nástroj manipulate location	52
6.5	Path editor	53
6.6	PTP pohyb	53
6.7	LIN pohyb	54
6.8	CIRC pohyb	54
6.9	Vizuální zobrazení parametru circ type	55
6.10	SPL pohyb	55
6.11	Robot configuration blok	56
6.12	Ukázka OLP příkazu	57
6.13	Aktivitní diagram operací	58
6.14	Operace profily	59
6.15	Operace světlo	60
6.16	Aktivitní diagram programu	61
7.1	Robot program viewer	63

Seznam tabulek

2.1	Prodejní verze Visual Components	18
3.1	Parametry KR 3 R540 [27]	30
3.2	Kartézské souřadnice TCP bodu [27]	31
3.3	Parametry KR C4 compact [27]	32
3.4	Skupiny uživatelů [27]	38
3.5	Pracovní režimy [27]	38
4.1	Nadefinované báze v Process Simulate	43
4.2	Nadefinované nástroje v Process Simulate	43
6.1	Použité vstupní a výstupní signály	59
7.1	Nástroje v PS a upraven v kontroléru robotu	64

Kapitola 1

Úvod

Diplomová práce se zaměřuje na analýzu nástrojů pro offline programování robotů. Práce dále popisuje vytvoření digitálního dvojčete laboratorní úlohy s KUKA KR R3 450 šestiosým robotickým ramenem v prostředí Tecnomatix Process Simulate. Dále je vytvořena a realizována úloha, která ověří princip offline programování. Programování robotické úlohy je realizováno v digitálním dvojčeti a následně je exportováno a testováno na reálném zařízení.

V první části diplomové práce je teoreticky rozebrána problematika offline programování robotických ramen. Popsány jsou výhody a nevýhody offline programování. Poté jsou vypsány jednotlivé nástroje, které byly nalezeny v rámci analýzy. Celkem bylo nalezeno třináct softwarů, které se zabývají problematikou offline programování. U těchto nástrojů jsou vypsány jejich základní parametry, které podporují.

Další část práce popisuje laboratorní instalaci KUKA ready2_educate, která bude převedena do digitální podoby. V popisu je popsáno robotické rameno KR 3 R540, řídicí jednotka, chapadlo, smardPad a uživatelské rozhraní.

K tvorbě digitálního dvojčete byl zvolen software Tecnomatix Process Simulate. V této části práce je popsáno nastavení softwaru pro správné exportování vytvořených operací pro KRC kontrolér. Popsán je postup nastavení Process Simulate pro krl jazyk, vytvoření kinematiky chapadla a robotu, vytvoření bodů na montážní desce, definování nástrojů a bází a import modelů, které byly nezbytné pro vytvoření projektu.

V práci je dopodrobna popsáno vytváření úlohy pro offline programování. V této kapitole bylo řešeno zjednodušení digitálního modelu světloometu, který byl vytvořen pomocí 3D skeneru. Dále je pro světlomet navržena konstrukce, která jej bude v buňce držet. Konstrukce se skládá z dílů vytvořené pomocí metody 3D tisku a hliníkových profilů.

Podrobně je popsáno programování a vytváření robotických operací v Tecnomatix Process Simulate a export finálního programu do reálného kontroléru. Popsány jsou jednotlivé parametry, které se definují u bodů. Dále je vytvořen hlavní program robotu, ze kterého jsou volány podle vstupního signálu z HMI panelu dvě operace. První vytvořená operace najíždí nad montážní otvor v profilech.

Postupně sjede nad všechny čtyři profily. Druhá operace objede hranu světlometu pomocí kostky s hrotem.

Poslední část diplomové práce popisuje nástroj v Tecnomatix Process Simulate, který zobrazí vytvořené operace přímo v jazyce krl. Poté jsou popsány vyexportované soubory a jak byly nahrány do reálné řídicí jednotky robotu. V poslední části této kapitoly je popsán postup sjednocení obou pracovišť, aby digitální model odpovídal reálnému prostředí.

Kapitola 2

Offline programování robotů

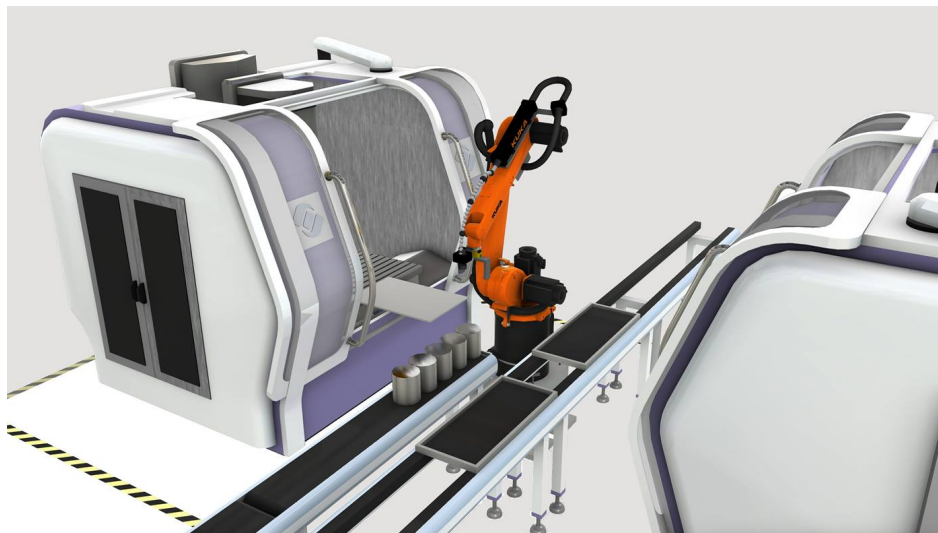
Robotické aplikace ve výrobě vedou ke zkrácení doby a zlepšení kvality. V průmyslovém prostředí existují dvě hlavní metody programování robotů online programování a offline programování (OLP). Online programování nevyžaduje žádný další hardware a software kromě těch, které se používají pro výrobní proces. Vygenerovaný program je však velmi nepružný. Jestliže se provádí úpravy do již stávajícího procesu je potřeba zastavit výrobu, nebo daný proces. [1] [2]

Offline programování robotu pomocí CAD softwarů má vytvořit vizuální představení robotu při plnění jeho úkolu a ve fázi plánování eliminovat jeho problémy s dosahem, přístupností, kolizí, načítáním, hledáním ideálních trajektorií, zrychlováním procesů, atd. Programovat robotický systém lze pomocí komerčního OLP softwaru, který je dodán výrobcem OLP, nebo jinou firmou zabývající se touto problematikou. Další nesmírnou výhodou OLP je opakovatelnost použití kódu, flexibilita, menší prostoje během programování systému. OLP je obvykle ziskovější ve složitých aplikacích, jako je svařování, ořezávání, řezání laserem, odstraňování otřepů, termické stříkání, lakování, nanášení lepidla a další. OLP se nevyplácí u jednodušších aplikací jako je paletizace, balení, pick and place, atd.[2]

2.1 KUKA.sim

Simulační software pro efektivní offline programování, který je vytvořen pouze pro průmyslové roboty od výrobce KUKA. Kuka.sim nabízí vysokou efektivitu při offline programování, úsporu času, kontrolu kolizí mezi roboty a dispozicemi buněk. KUKA.sim nabízí možnost analýzy doby cyklu programů. Dále import již vytvořených programů a jejich úpravu. Program nabízí import CAD dat z formátů CATIA v5 a V6, JT, STEP a mnoho dalších. Rozsáhlá online knihovna s aktuálně dostupnými modely robotů, chapadel, dopravníků a mnoho dalších. Z KUKA.sim lze exportovat 2D výkresy vytvořených projektů. Další možnost, kterou Kuka.Sim nabízí, je využití virtuální reality. Verze KUKA.sim PRO 3.0 má v balíčku i software KUKA.OfficeLite. KUKA.Sim Pro nabízí realtime

připojení k KUKA.OfficeLite. Možnost simulovat vstupní a výstupní signály. Prostředí KUKA.Sim lze vidět na obrázku 2.1. [3]



Obrázek 2.1: Prostředí KUKA.Sim [3]

2.1.1 KUKA.OfficeLite

KUKA.OfficeLite je virtuální ovladač robotů KUKA. Pomocí programovacího systému je možno na každém počítači offline vytvářet a optimalizovat programy. Hotové programy jsou přímo přenositelné do robotu a zajišťují okamžitou produktivitu, jelikož je téměř identický se systémovým softwarem KUKA řídicího systému KR C4. KUKA.Office lite zajišťuje kontrolu syntaxe KRL pomocí disponibilního kompilátoru a překladače. Prostředí KUKA.Sim lze připojit k programu Tecnomatix Process Simulate od v 16.0. Prostředí lze vidět na obrázku 2.2. [4]

2.2 Tecnomatix

Tecnomatix je seskupení produktů, které se zabývají kompletně digitalizovanou výrobou, které pomáhají digitalizovat výrobu. Se SW Tecnomatix lze dosáhnout synchronizaci mezi produktovým inženýrstvím, výrobním inženýrstvím, výrobou a servisními operacemi. [5]

V rámci rodiny Tecnomatix jsou dva SW, které se zabývají offline programováním průmyslových robotů a to Process Simulate a Robot Expert. Tyto dva softwary jsou plně identické, ale rozdíl je v jejich licencování. Process Simulate může být ve formě serverové licence, kde celý SW běží pomocí platformy Teamcenter na severu, a nebo samostatná licence, která běží na PC. Robot Expert může pouze běžet samostatně na PC, nelze jej v rámci serverového řešení Teamcenter používat. [5]

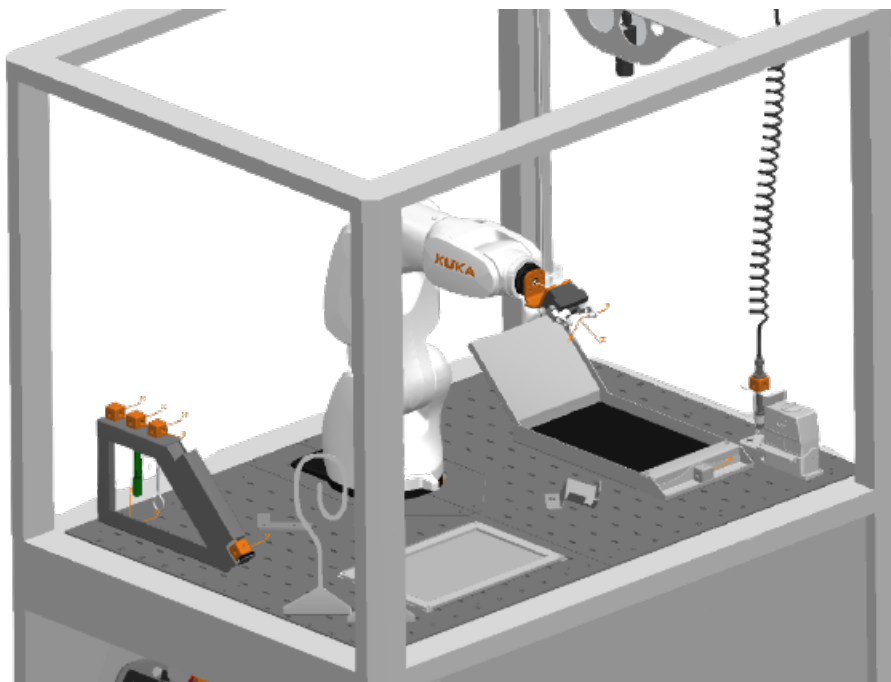


Obrázek 2.2: Prostředí KUKA.OfficeLite [4]

2.2.1 Process Simulate

Process Simulate je digitální simulační řešení, které slouží k ověření výrobního procesu s využitím robotů v 3D prostředí. Process Simulate je nástroj, který řeší několik úrovní simulace robotů a vývoje pracovních stanic, od stanic s jedním robotem až po kompletní výrobní linky a pracovní zóny. Pomocí tohoto nástroje lze zlepšit komunikaci a koordinaci mezi výrobními disciplínami a umožnit tak chytřejší rozhodování. Software Process Simulate nabízí možnost offline programování robotických ramen. Při tvorbě digitálních dvojčat disponuje mnoha nástroji pro jejich přesné zpracování, např. tvorbu dopravníků, tvorbu materiálů potřebných pro výrobu, práci s mnoha druhy senzorů, tvorbu kinematiky, tvoření logických funkcí pro jednotlivé objekty a další. Software Process Simulate nabízí možnost simulování lidských pohybů při vykonávání jejich práce, tyto data lze analyzovat a upravit jejich pracoviště tak, aby byla jejich práce méně náročná a došlo například i k zrychlení výroby. K virtuálnímu zprovoznění nabízí Process Simulate připojení signálu k reálnému PLC s využitím OPC serverů, a nebo připojení k simulovanému PLC. Nejnovější verze Process Simulate nabízí simulaci AGV, díky které lze detekovat kolize, upevňovat nástroje na konstrukci vozidel a plánovat efektivnější trasy. Prostředí Process Simulate lze vidět na obrázku 2.3. [6] [7]

Ke generovaným cestám lze přidat podrobné informace, včetně atributů pohybu a procesu pro vytvoření kompletních programů stažených do skutečného kontroléru. Process Simulate nabízí automatické generování trajektorií na základě vytvořených křivek. Tvorbu svařovacích operací, svařovacích nástrojů a diagnostiku svárů. Process Simulate dokáže kontrolovat dosah robotu, umístit robot do prostoru podle potřeby a lze provádět jednoduché úpravy v již vytvořených operacích, které byly do programu naimportovány. Samozřejmostí je možnost nahrávat vytvořené programy do robotů. Díky využití virtuálních kontrolérů robotů je dosažena maximální přesnost simulace, přičemž program robotu je pak generován bez nutnosti úprav. Process Simulate nabízí kontroléry



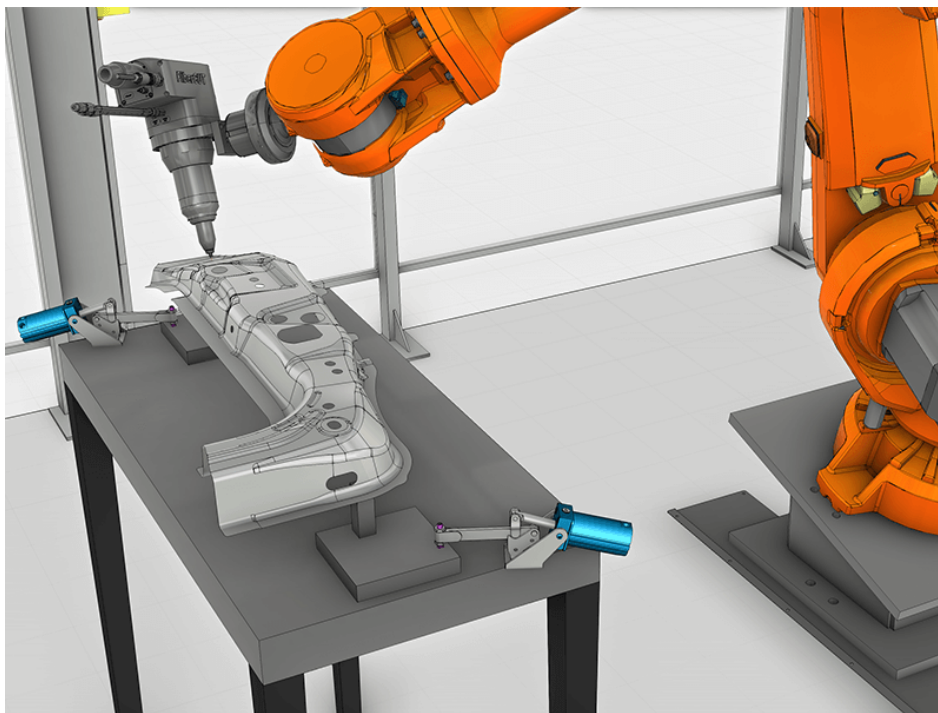
Obrázek 2.3: Prostředí Tecnomatix Process Simulate

od mnoha výrobců průmyslových robotů např. KUKA, ABB, Fanuc, Mitsubishi, a mnoho dalších. [6] [7]

2.3 OCTOPUZ

Kanadská firma OCTOPUZ Inc. se 15 let specializuje na design, vývoj, implementaci a přizpůsobení softwaru pro offline programování průmyslových robotů. Firma nabízí vývojové prostředí OCTOPUZ 3.0. Software OCTOPUZ umožňuje automatizaci úloh, které jsou pro pracovníky opakující se, zdlouhavé, nebezpečné nebo nevhodné, a to díky snadnému programování a přeprogramování průmyslových robotů. Program nabízí programovat aplikace i s více roboty. Nabízí tvorbu svařovacích, nástřikových, obráběcích, řezacích a aditivních operací. U svařovacích aplikací software nabízí implementaci mnoha druhů svařovacích postupů, jako jsou více vrstevové svary, bodové svařování, snímání dotyku, sledování svaru a další. OCTOPUZ dokáže automaticky napočítat optimální místa pro snímání dotyku, nebo ručně vytvářet jedno, dvou nebo tří bodové snímání dotykem. Simuluje a analyzuje tloušťku naneseného nástřiku na povrch. Podpora výrobců robotických ramen ABB, FANUC, KUKA, YASKAWA a mnoho dalších. Prostředí lze vidět na obrázku 2.4. [8]

Dále OCTOPUZ nabízí možnost výrobcům a integrátorům vytvářet ve virtuálním prostředí přesně své robotické buňky. Jednotlivé komponenty lze vkládat do prostředí z rozsáhlé knihovny. Virtuální prostředí nabízí testování složitých mechanických dílů, optimalizaci operací a manipulací s díly. Nastavit komunikaci mezi vstupy a výstupy jednotlivých komponent. [8]



Obrázek 2.4: Prostředí OCTOPUZ [8]

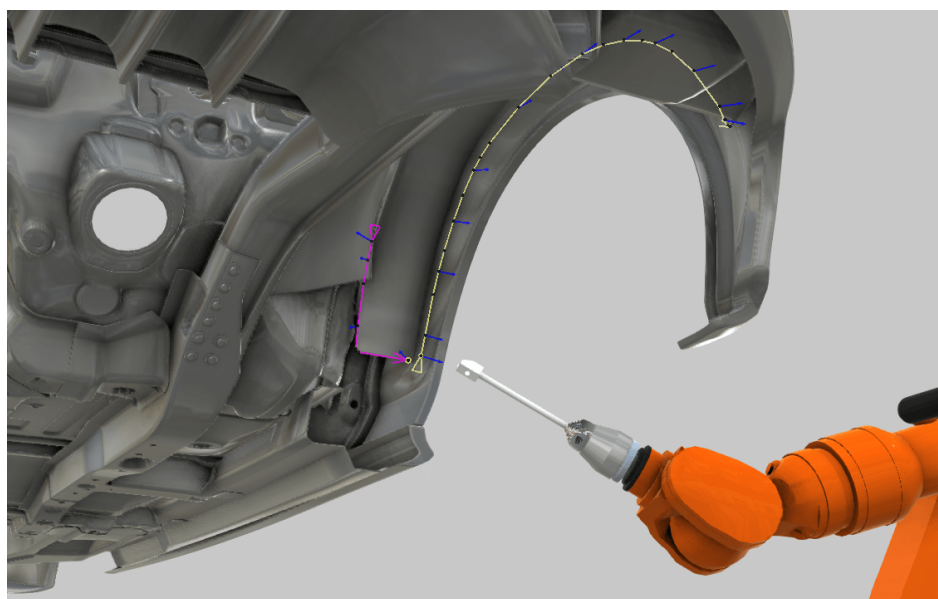
2.4 Visual Components

Visual Components je předním vývojářem softwaru a řešení pro simulaci 3D výroby. Společnost byla založena v roce 1999 týmem odborníků na simulace. Platforma Visual Components byla navržena tak, aby podporovala pokročilé aplikace 3D simulace výroby. Vše pohání opensource fyzikální engine NVIDIA PhysX, v simulačním prostředí jsou objekty a funkce ovlivněny gravitací, vlastnostmi materiálů a kolizí. Software dále nabízí možnost tvorby vlastních API s využitím skriptovacího programovacího jazyka Python. Visual Components je nabízen ve třech verzích a to Essentials, Professional a Premium, viz. tabulka 2.1. [9]

Z pohledu OLP nabízí Visual Components programování robotických úloh ve virtuální prostředí. Nabízí nástroje pro tvorbu křivek, které jsou schopné automaticky vytvořit kód pro robotickou operaci, dle vytvořené křivky. Visual Components při importu CAD modelů analyzuje model a vytvoří strukturovaná data povrchů. Import dat z katalogu, který obsahuje mnoho předem vytvořených komponent. Umožňuje rychlý návrh a tvorbu robotických buněk. Pro výrobce robotických ramen Universal Robot je vyvinut speciální plugin pro připojení ke kontroléru a lze spouštět simulovaný robot s realistickými pohyby. Podobný plugin je vytvořen i pro roboty Stäubli, ve kterém lze vytvářet a upravovat pozice robotických programů a ověřit činnost programů a doby cyklů. Produkty pro OPL, které vyvinuli jejich partneři Defoli a OCTOPUZ. Prostředí Visula Components lze vidět na obrázku 2.5. [10]

Tabulka 2.1: Prodejní verze Visual Components

Verze Visual Components			
Verze	ESENTIAL	PROFESSIONAL	PREMIUM
Konfigurace rozložení	Ano	Ano	Ano
Procesní modelování	Ano	Ano	Ano
CAD Kompatibilita	Ano	Ano	Ano
Projektové výstupy	Ano	Ano	Ano
Jednoduchá robotika	Ano	Ano	Ano
Modelování komponent	Ne	Ano	Ano
Pokročilá robotika	Ne	Ne	Ano
Interaktivní VR	Ne	Ne	Ano



Obrázek 2.5: Prostředí Visual Components [9]

2.5 Delfoi

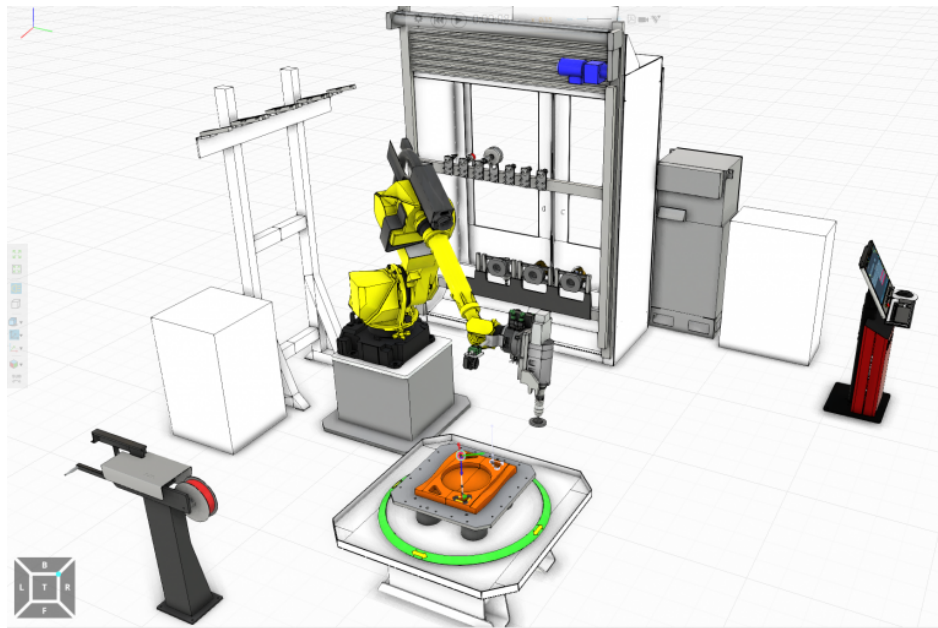
Delfoi je firma založená v roce 1990, která se od začátku zabývá problematikou OLP. Delfoi v dnešní době nabízí hodně produktů v rámci Industry 4.0. Nabízí tři základní zaměření v rámci OLP a to Delfoi Planner, Delfoi Robotics a Delfoi Lean. Delfoi Planner se zaměřuje na plánování a optimalizaci výroby. Delfoi Lean je projektový model, který umožňuje zlepšení produktivity, bez přidání výrobních zdrojů. Delfoi Robotics nabízí vývojová prostředí Delfoi ARC, Delfoi CUT a Delfoi PAINT, která pokrývají komplexní oblasti aplikací průmyslové robotiky. Každý z těchto softwarů se zaměřuje na odlišnou problematiku, ale mají společně rozsáhlé knihovny plné komponent pro modelování buněk (dopravní pásy, bezpečnostní bariéry, světelné závory, detekci kolizí, detekce porušení limitů stroje a mnoho dalších). [11, 12]

2.5.1 Delfoi Arc

Delfoi Arc nabízí rozsáhlé funkce pro tvorbu svařecích operací a jejich snadnou úpravu, kalibraci nástrojů a integrovanou kontrolu kvality. Podpora spolupráce více robotů. Delfoi , správa a vlastnosti procesů svařování, např. svařovací napětí, rychlost svařování a sledování svaru. Software je schopný hledat pozice pro svařování manuálně nebo automaticky. Softwarová sada Delfoi Robotics využívá osvědčenou platformu simulační technologie poskytovanou Visual Components. Software podporuje všechny hlavní značky robotů, jako jsou KUKA, ABB, FANUC, Panasonic, atd.[13]

2.5.2 Delfoi Auto Arc

Delfoi Auto ARC je nový software pro automatické OLP, kde je zapotřebí roboticky provádět obloukové svařování otevřených ocelových konstrukcí, jako jsou konstrukční nosníky, vazníky nebo lodní bloky. Tento software je extrémně rychlý při generování plně funkčních a bezkolizních robotických programů. Tento SW je vhodný pro jedinečnou výrobu a umožňuje robotizované svařování bez nutnosti programování přerušení pro tradičně manuální pracovní procesy. Dokáže automaticky detekovat všechny vodorovné místa vhodné pro svar.[14]



Obrázek 2.6: Prostředí Delfoi CUT [15]

2.5.3 Delfoi Cut

Delfoi CUT (obrázek 2.6) lze využít v široké škále aplikací, jako je robotické odstraňování otřepů, laserové řezání a plazmové řezání. Dále nabízí automatický import drah nástrojů CNC z programů

CAD/CAM do simulačních modelů. Podporuje automatizovanou, neomezenou správu všech externích os robotů např. portálové a lineární dráhy nebo víceosé polohovací zařízení. [15]

2.5.4 Delfoi Paint

Delfoi PAINT je software, který je schopen efektivně využívat vlastnosti 3D CAD modelu, který má být malován, a informace o topologii povrchu. Automaticky počítá trajektorie na základě charakteristik povrchu. Software dokáže analyzovat tloušťku naneseného nástřiku, tvořit přesnou simulaci pomocí variabilních parametrů, tolerancí a přesnosti. Volit výběr a tvorbu různých barevných přechodů pro simulaci. Dále nabízí správu robotických lakovacích pistolí a výměny pistolí, simulaci pohybu hadic, které slouží k dodávce barvy do pistole. Delfoi PAINT umí číst geometrická data ze 3D skeneru a aplikovat je na simulační model a využívat je při vytváření trajektorií. Dokáže automaticky počítat rychlosti lakování s ohledem na tloušťku nástřiku. [16]

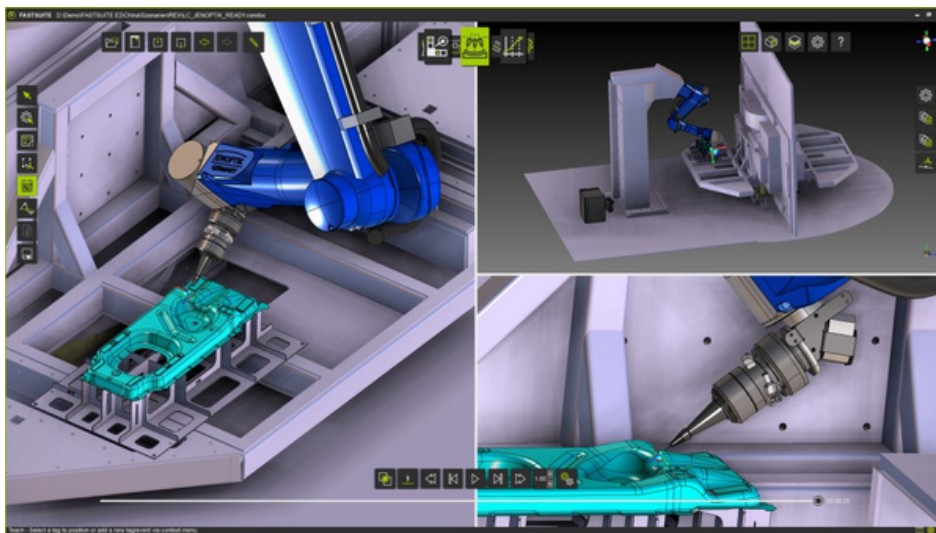
2.6 FASTSUITE

FASTSUITE OLP je založeno na softwarové architektuře a konceptu, který umožňuje uživatelům snadno programovat roboty nebo obráběcí stroje s využitím procesní geometrie definované na obrobcích a dostupných technologických balíčcích. FASTSUITE nabízí dva produkty FASTSUITE pro V5, který se integruje do CATIA/DELMIA V5 (DELMIA V5 je software, který se specializuje na digitální výrobu a simulaci výroby). Druhý SW, který firma nabízí FASTSITE Edition 2 je nezávislá platforma, která slouží k tvorbě digitální továrny. [17]

FASTSUITE Edition 2 automaticky vybere odpovídající technologický balíček na základě nástroje namontovaného na robotu. Různé balíčky technologií poskytnou GUI přizpůsobené přímo pro konkrétní technologii a vypočítají dráhu nástroje způsobem, který je již optimalizován pro danou technologii. Podporované technologie obloukového svařování nebo laserového řezání obsahují již všechny specifické procesy příkazy pro ovladač. FASTSUITE nabízí programování robotických operací jako bodové a obloukové svařování, laserové řezání, nástřik a pick&place. Při tvorbě pick&place operací je podpora více TCP, události k ovládání operací uchopení a uvolnění, virtuální určení pozic robotů a kontrola poloh robotu prostřednictvím panelů. Prostředí lze vidět na obrázku 2.7. [17]

Laserové svařování nabízí speciální integraci technologie, která byla dokonale přizpůsobena robotům TruLaser Robot 5020, TRUMPF TruLaser Weld 5000 nebo Amada FLW 4000 M3 a dalším systémům, nabízí FASTSUITE optimální podporu pro efektivní používání, provoz systému a získání vysoce kvalitních laserových svařovacích programů. Softwarová podpora výběru správných WPS z přizpůsobitelné databáze schválených parametrů svaru. [17]

FASTSUITE Edition 2 umožňuje výrobcům vytvářet, simulovat a analyzovat nástřikové aplikace a vylepšovat vzory stříkání, pokrytí povrchu, prodloužení rozprašovacího kužele a rychlosti nanášení. Simulace umožňuje optimalizaci trajektorií robotů tak, aby se rozdělilo správné množství stříkaného



Obrázek 2.7: Prostředí Fastsuite [17]

materiálu a dosáhlo se rovnoměrného usazování po povrchu, což zajišťuje pokrytí a zamezení plýtvání materiálu. [17]

2.7 CIROS Studio

CIROS Studio (obrázek 2.8) je software pro tvorbu digitálních dvojčat v 3D prostředí. V programu CIROS Studio uživatelé modelují rozvržení a procesy, simulují robotické pracovní buňky a automatizované výrobní závody a vizualizují složité sekvence. CIROS Studio spojuje plánování, design, elektrických rozvodů, vývoj procesů, uvádění zařízení do provozu, prodej a marketing. Dále program nabízí plánování a optimalizaci času cyklů, automatický výpočet trajektorií pro zpracování povrchů obrobků a i následné ruční zpracování. Při virtuálním uvedení od provozu lze využít skutečné PLC. CIROS Studio se například hodí do odvětví automobilového, automatizace, výroby strojů, strojírenství a obrábění. CIROS Studio nabízí k základní aplikaci SW balíčky, které rozšiřují základní SW o problematiku, kterou balíček obsahuje, např. import modelů.[18]

CIROS Studio nabízí v problematice offline programování rozsáhlou knihovnu průmyslových robotů od 17 výrobců. Podporuje spolupráci více kooperací průmyslových robotů. Dále program nabízí snadné modelování uživatelsky definované kinematiky. Balíček robotiky nabízí virtuální kontrolér robotu, virtuální učení pohybů, správu TCP bodu, vizualizaci pracovního prostoru, analýzu dosažitelnosti v závislosti na kinematice, rotační a lineární přídatné osy. Dále nabízí detekci kolizí pro libovolné prvky (roboty, chapadla, uchycené prvky v chapadlech a další). Kolize lze zapisovat do protokolu, každá kolize generuje zprávu s časovým razítkem pro pozdější analýzu. Další robotické balíčky se zaměřují na výrobce průmyslových robotických ramen, např. ABB s programovacím jazykem RAPID a knihovnou modelů robotů, KUKA s programovacím jazykem KRL a knihovnou



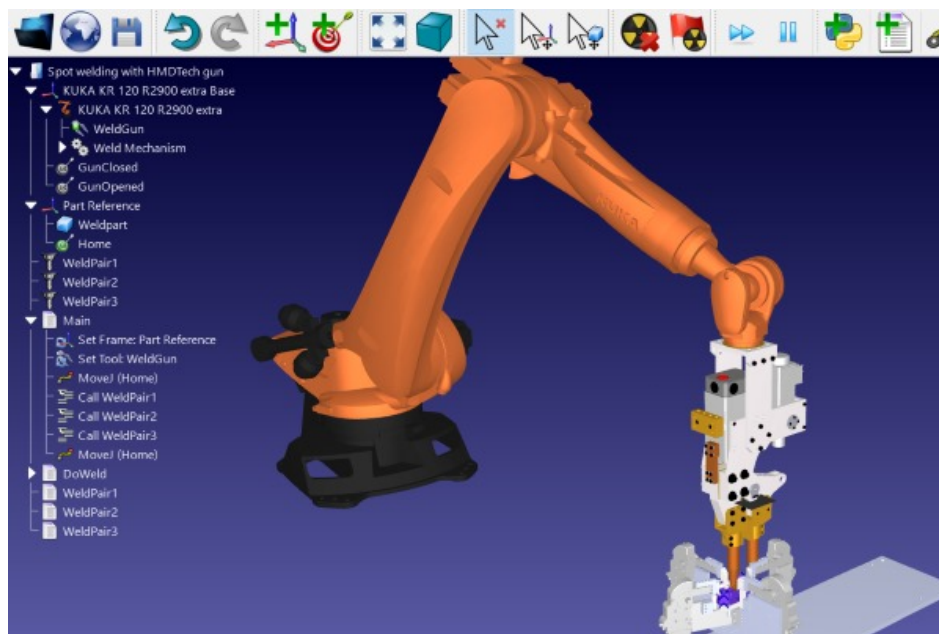
Obrázek 2.8: Prostředí CIROS Studio [18]

modelů robotů a další. Další balíčky se zaměřují na mechanismy, import a práci s modely, uvádění do provozu, tvorba videí, virtuální osoby, virtuální reality a mnoho dalších. [18]

2.8 RobotDK

Firma RobotDK nabízí stejně jmenovaný SW, firma vznikla jak spinoff v roce 2015 v Kanadě. RobotDK nabízí rozsáhlou knihovnu průmyslových robotických ramen od 40 různých výrobců (ABB, KUKA, Fanuc, Universal Robots a další). Knihovna podporovaných průmyslových robotických ramen je dostupná online na stránkách výrobce softwaru. RobotDK nabízí konfiguraci až tří externích os, jako jsou lineární kolejnice a otočné talíře. Další funkcí softwaru je možnost modelování a synchronizování dalších os. Podporovaný import 3D modelů a vlastních nástrojů, importované formáty STL, STEP a další. Simulování drah robotů nabízí možnost vytvářet obráběcí, pick&place, nástřikové, vrtací, laserové, svařovací a 3D tiskové operace. RobotDK API umožňuje programovat libovolného robotu pomocí jednoho z podporovaných programovacích jazyků, jako je C#, Python nebo C++. Opakované úkoly můžete také automatizovat pomocí rozhraní RobotDK API. RobotDK podporuje virtuální realitu. Prostředí lze vidět na obrázku 2.9. [19]

RobotDK je nabízen ve čtyřech verzích. Ve všech prodejních verzích je dostupná celá knihovna průmyslových robotických ramen. První verze je zdarma ke stažení se všemi funkcemi po dobu 30 dní. Omezené je generování programu na 50 řádků kódu. Druhá "Education" verze je dodávána pouze do akademického prostředí. Tato licence nabízí všechny funkce offline programování, robotické obrábění, 3D tisk, neomezené generování programu a přístup ke všem kontrolérům. Cena této licence je 145 euro. Třetí "Professional" verze nabízí navíc oproti education licenci možnost simulovat kooperaci více robotů najednou, podporu externích os a synchronizaci až 12 os. Cena této licence je



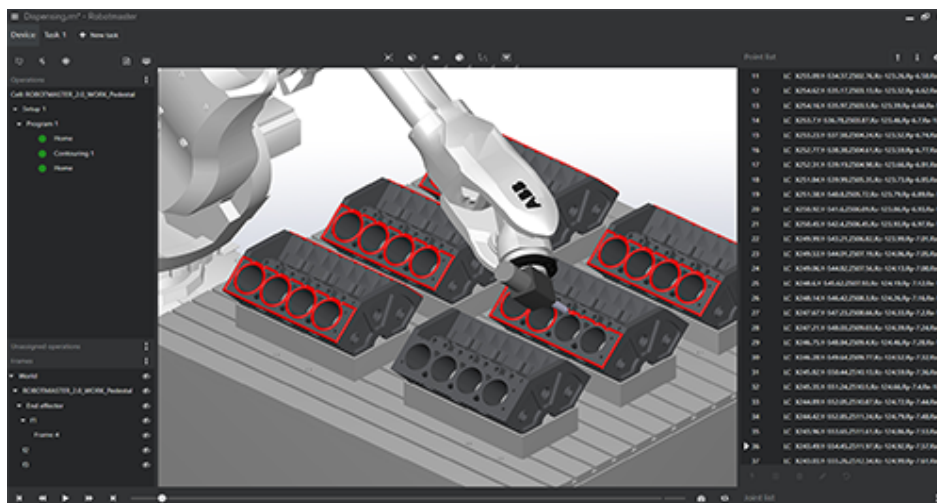
Obrázek 2.9: Prostředí RobotDK [19]

2995 euro. Poslední verze calibration&performance testing, nabízí navíc oproti professional kalibraci a testování výkonu. Cena této licence je na vyžádání.[19]

2.9 Robotmaster

Robotmaster vznikl v roce 2001 jako myšlenka, která se chtěla oprostit od klasického programování robotů a využít možnost offline programování. Nyní Robotmaster patří pod společnost Hypertherm. Software poskytuje nástroj pro programování, vizualizaci a optimalizaci robotických procesů. V programu Robotmaster lze parametrizovat rotaci nástroje, náklon nástroje a polohy kolejnice a rotace. V prostředí lze vytvářet aplikace v oblastech ořezávání, řezání, robotické obrábění, odstraňování otřepů, svařování, leštění, broušení, lakování a další. Integrace parametrů specifických pro proces uživatele je zjednodušena pomocí upravitelných obrazovek, které definují interakci, terminologii a nastavení řízení jedinečné pro aplikaci. Robotmaster podporuje značky robotů jako je ABB, KUKA, FANUC, YASKAWA a mnoho dalších. [20]

Robotmaster (obrázek 2.10) nabízí při programování průmyslového robotu, možnost orientovat nástroj automaticky v závislosti na optimalizaci. Dále nabízí schopnost zpracovat vstupy a výstupy, simulovat a programovat více nástrojů pro roboty. Robotmaster podporuje snadnou tvorbu aditivních výrobních procesů, jako je 3D tisk, laserové tavení kovů, laserové nanášení kovů a další. [20]



Obrázek 2.10: Prostředí Robotmaster [20]

2.10 AUTOMAPPPS

Společnost Convergent Information Technologies se sídlem v Rakousku nabízí na trh produkt AUTOMAPPPS, který je souhrnem nástrojů pro programování průmyslových robotů. Softwarý řeší problematiku offline programování, bin pickingu, automatického programování robotů a robotické moduly.[21]

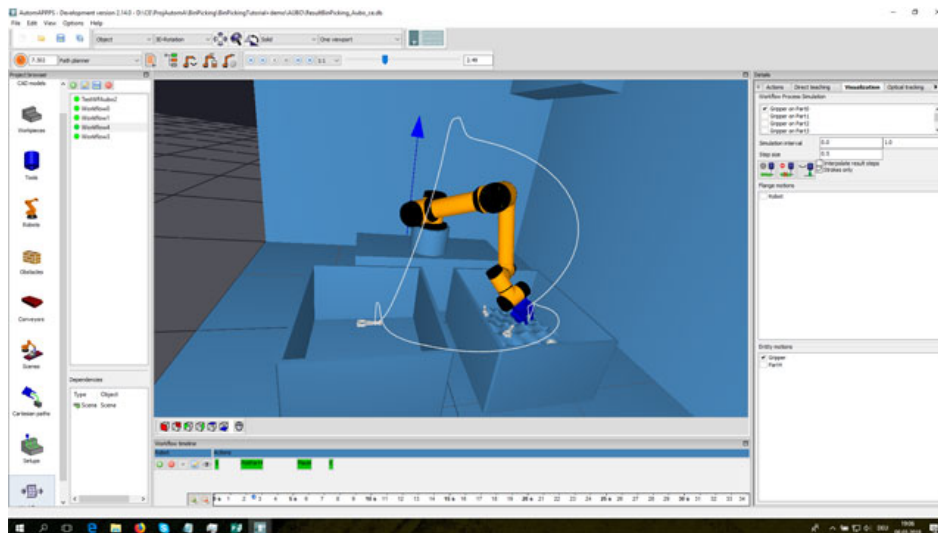
2.10.1 AUTOMAPPPS - OLP

Software pro programování robotů zahrnuje rychlé a snadné OLP a automatické generování kódu. Programovací nástroj AUTOMAPPPS nabízí plánování pohybu robotu a podporuje více než 15 výrobců robotů (ABB, KUKA, Omron, Universal Robots, a další). Podpora formátů JT, STEP, STL a další. Prostředí AUTOMAPPPS lze vidět na obrázku 2.11. [21]

SW obsahuje programování s nízkým a nebo žádným kódem, kromě toho je vytvořeno automatické předcházení kolizím a automatické plánování. AUTOMAPPPS nabízí automatické určování polohy robotického ramene v buňce a ověřování dostatečné velikosti buňky pro vytvořené operace. AUTOMAPPPS má plnou podporu pro svařování, obrábění, pískování, řezání, aditivní výrobu a další. Software podporuje koordinaci více robotů se stejným pracovním prostorem. Dále nabízí automatickou optimalizace sekvencí pomocných os. [21]

2.11 FAMOS

Carat robotické inovace GmbH FAMOS robotic vyvíjí softwarový systém FAMOS (obrázek 2.12) pro offline programování, simulaci a optimalizaci procesů průmyslových robotů. CAD modely lze importovat v různých formátech a snadno je použít pro generování trajektorií. Cesty lze vytvářet



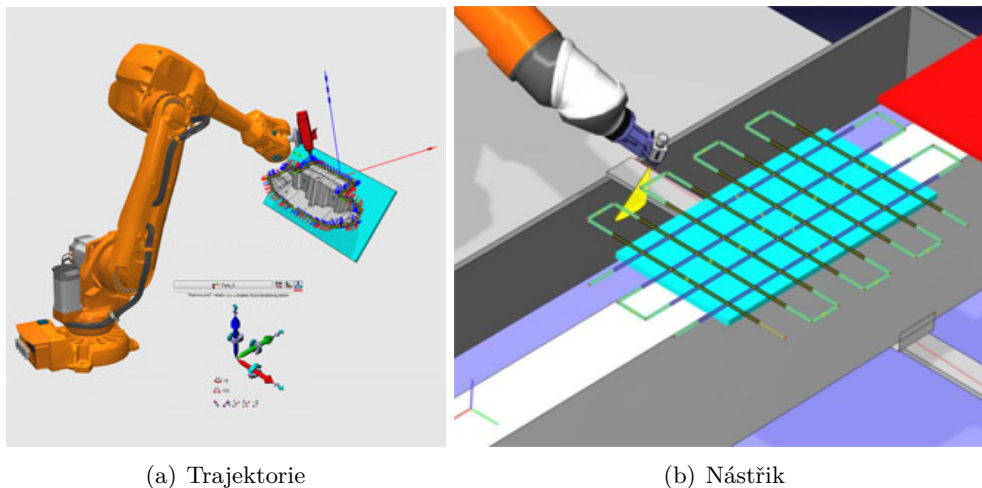
Obrázek 2.11: Prostředí AUTOMAPPPS [22]

například podél souvislých křivek a hran, na plochách a průsečících. Všechny souřadné systémy nástrojů a základen lze přesouvat a otáčet relativně ke každému jinému souřadnicovému systému v projektu. Hodnoty lze měnit postupně nebo je zadat přímo. Orientace se zobrazují automaticky ve formátu příslušného robotu. FAMOS podporuje mít ve stejném simulačním projektu použití několik robotů, kteří mohou být od různých výrobců. Počet robotů v projektu je omezen pouze výkonem počítače. Roboty v projektu lze vyměnit za jiné i v běžící simulaci, vyměňovat lze i roboty od jiných výrobců. Program podporuje hlídání kolizí mezi roboty a pracovním prostorem. Knihovna robotů nabízí okolo 500 modelů a podpora značek robotů ABB, KUKA, FANUC a další. Firma nabízí možnost vytvoření modelu, když se nenachází v knihovně. FAMOS lze využít v aplikacích jako je leštění, broušení, lepení, řezání a další.[23]

FAMOS nabízí psaní skriptů, které umožňují vytvářet vlastní funkce. Pro programování skriptů se využívá jazyk pascal. Pomocí skriptů lze vytvářet automatické úkoly, které se neustále opakují. Tímto je myšleno tvoření duplicitních cest a generování nových cest. Import a export bodů tras z textového souboru ve vlastním formátu. [23]

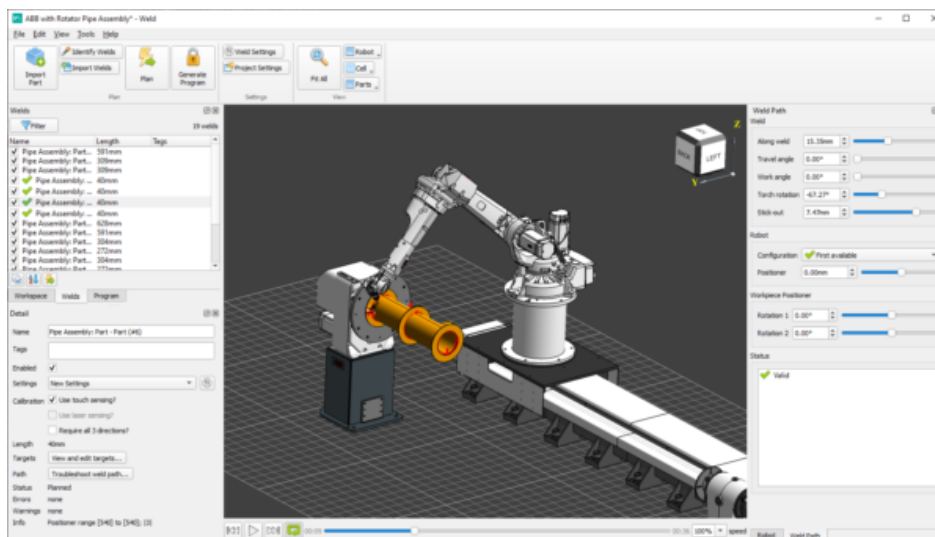
2.12 Verbotics Weld

Společnost Verbotics řeší nabízí inteligentní software pro generování robotických svařovacích programů přímo z informací CAD modelů (obrázek 2.13). Software pomocí algoritmů z importovaných modelů navrhnout svary, generování není ovlivněno tvarem a složitostí modelu. Nevrhnuté svary automaticky parametrizují úhly natočení svařovacího nástroje. Svary lze také importovat z tabulek CSV. Vygenerované svary se dají jednoduše modifikovat. Dráhy svařovacího nástroje jsou automaticky plánovány a optimalizovány. Software podporuje externí osy (portály a rotační) a 7osé roboty.



Obrázek 2.12: Prostředí FAMOS [23]

Robotické operace jsou optimalizovány pro nejkratší dobu cyklu. Verbotics Weld nabízí automatickou dotykovou nebo laserovou kalibraci. Software v současné době podporuje offli programování pro robotická ramena od výrobců ABB, FANUC a Yaskawa Motoman's. Software Verbotics Weld je na webových stránkách výrobce nabízen ve zkušební verzi, která je zdarma ke stažení. [24]

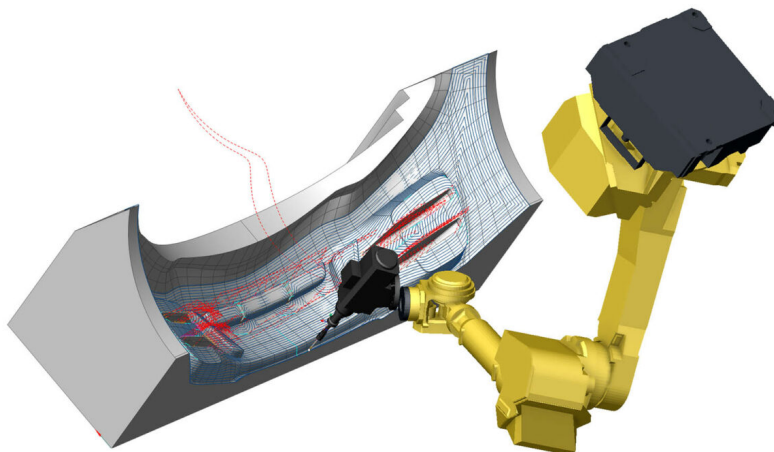


Obrázek 2.13: Prostředí Verbotics Weld [24]

2.13 SprutCAM Robot

SprutCAM Robot (obrázek 2.14) je CAD/CAM software pro offline CNC programování průmyslových robotů a software pro robotickou simulaci v jednom integrovaném balíčku vytvořený firmou

SPRUT Technology. Dráhy nástroje jsou počítány nativně v kódu pro šesti nebo více osé robotické aplikace. Software nabízí funkce pro detekci kolizí mezi roboty, vyhýbání se singularitám, detekci limitů pracovních zón. SprutCAM Robot podporuje současné řízení robotu a neomezený počet lineárních a rotačních polohovacích os.[25]



Obrázek 2.14: Prostředí SprutCAM Robot [25]

Knihovna, která je součástí softwaru zahrnuje širokou škálu hlavních výrobců robotů, např. Fanuc, KUKA, ABB, Nachi, atd. Knihovny také obsahují šablony, které umožňují rychlé tvoření uživatelských kinematik. V rámci obrábění obrobků nabízí software různé strategie obrábění obrobků pomocí nástrojů např. obrábění děr, hrubování rovinné a rotační, paralelní a mnoho dalších. Dále nabízí dokončovací dráhy nástrojů. SprutCAM Robot nabízí pokročilý modul pro simulaci obrábění, který obsahuje přesnou simulaci kinematiky robotů v reálném čase, simulaci odebrání materiálu, detekci mechanických kolizí, singularity robotu a zóny mimo dosah robotu a další. [25]

Kapitola 3

Laboratorní pracoviště

KUKA ready2_educate (obrázek 3.1) je tréninkovou buňkou pro školení obsluhy robotů. Je určena zejména pro školy, vysoké školy a firemní vzdělávací zařízení. Tréninková buňka představuje kompletní startovní balíček. Buňka je vybavena malým robotickým ramenem KR 3 AGILUS (KR 3 R540) a řídicím systémem KUKA KR C4. [26]

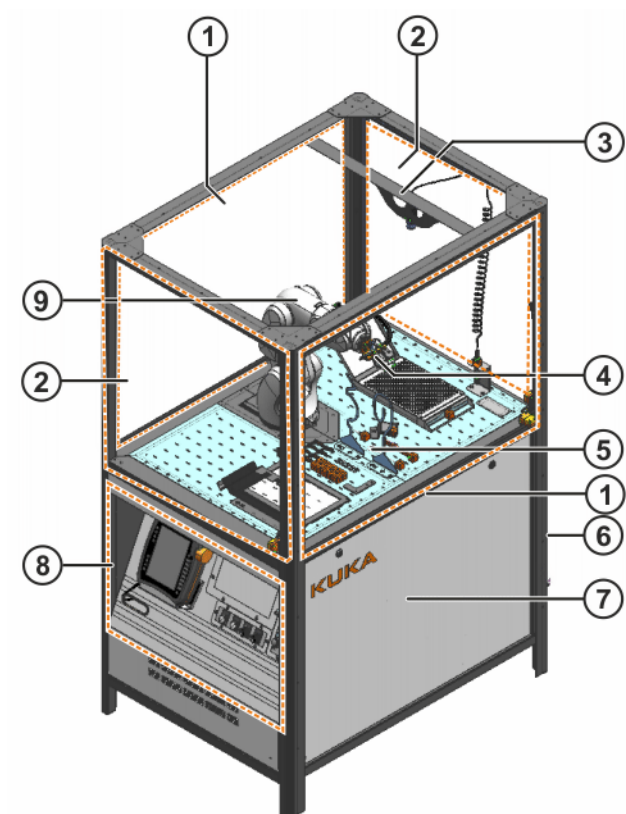
KUKA nabízí školící buňku ve více variantách a to ready2 educate basic, ready2 educate pro, ready2 educate advanced, ready2 educate pro vision, ready2 educate advanced vision, práce byla realizována na variantě ready2 educate advanced vision. [26]

Části robotické buňky jsou zobrazeny na obrázku 3.1, ① Zvedací dveře. ② Ochranné dveře. ③ Systém Vision. ④ Chapadlo. ⑤ Aplikační deska s prvky. ⑥ Dveře pro údržbu. ⑦ Boční stěna (i na druhé straně). ⑧ Ovládací deska. ⑨ Rokot KR 3 R450. [27]

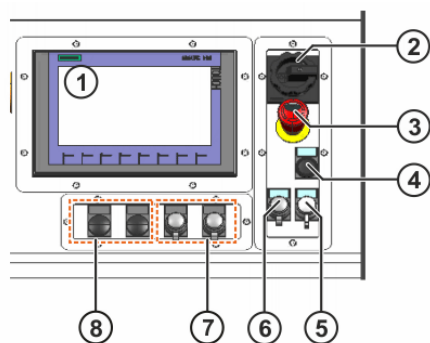
Ovládací část obsahuje ovládání, zobrazovací prvky a bezpečnostní systém nouzového zastavení. Jednotlivé prvky jsou zobrazeny na obrázku 3.2. ① Panel Simatic HMI Touch. ② Hlavní vypínač. ③ Tlačítko nouzového zastavení. ④ Kvitovat ochranu obsluhy. ⑤ Přípojka USB. ⑥ Přípojka KSI. ⑦ Přípojka Profinet. ⑧ Zaslupný kryt. Vlevo od HMI panelu se nachází místo pro uložení KUKA smartPADU.[27]

3.1 KR 3 R540

Robot KR 3 AGILUS je prostorově úsporný, má kompaktní velikost. Uvnitř robotu je uložený přívod elektrické energie a chráněné rozhraní na rameni umožňují flexibilní pohyby na minimálním prostoru. Ideální pro malé automatizační buňky, např. 600 x 600 mm. Robotické rameno KR 3 R450 je určen pro jmenovitou nosnost 2 kg, optimálně využívající výkonnost a dynamika robotu. Vyšší zatížení až do maximální hodnoty zatížení, lze použít při nižších intervalech. Robot umožňuje vysokou rychlost a tím tak zkrátit dobu cyklu na co nejkratší, při vykonávání nejrůznějších procesů. Robot je chráněn před nekontrolovaným elektrostatickým nabíjením nebo vybíjením a je přizpůsoben pro bezpečnou manipulaci s citlivými elektronickými součástkami. Robotické rameno je vodné využít pro procesy



Obrázek 3.1: KUKA ready2_educate [27]

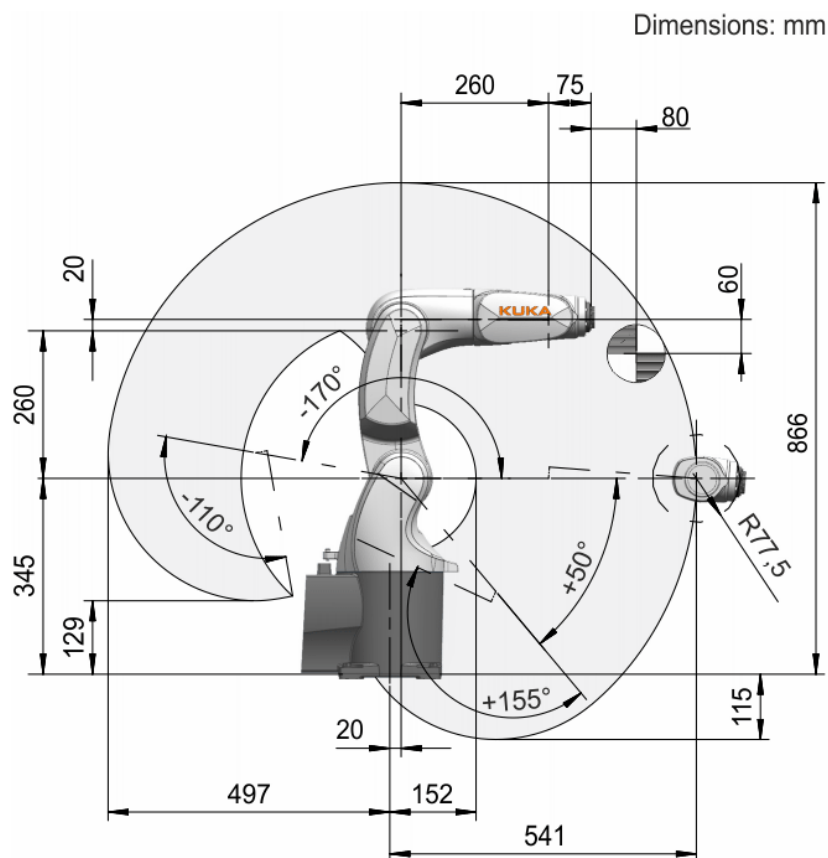


Obrázek 3.2: Ovládací a zobrazovací prvky buňky [27]

jako montáž, manipulace, paletizace, balení, měření, kontrolu, nános lepidla a lakování. Základní parametry jsou vypsány v tabulce 3.1 a pracovní zóny na obrázku 3.3.[28]

3.2 Chapadlo

Paralelní chapadlo se skládá z (obrázek 3.4). ① Přívodu energie. ② Držáku chapadla. ③ Dvě paralelní čelisti. Chapadlo slouží k přemísťování, uchopování a odkládání dílů. Čelisti mají dva



Obrázek 3.3: Pracovní zóny KR 3 R540 [28]

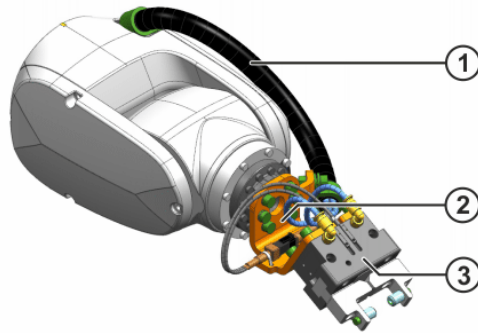
Tabulka 3.1: Parametry KR 3 R540 [27]

Základní technické údaje a rozsahy os	
Maximální dosah	541 mm
Maximální nosnost	3 Kg
Přesnost opakování polohy	$\pm 0,02$ mm
Počet os	6
Montážní poloha	podlaha, strop, stěna
Instalační plocha	179 x 179 mm
Hmotnost	26,5 kg
A1	$\pm 170^\circ$
A2	$-170^\circ / 50^\circ$
A3	$-110^\circ / 155^\circ$
A4	$\pm 175^\circ$
A5	$\pm 120^\circ$
A6	$\pm 350^\circ$

Tabulka 3.2: Kartézské souřadnice TCP bodu [27]

Kartézské souřadnice TCP bodu						
	X	Y	Z	A	B	C
TCP	+34.7 mm	0 mm	+109.3 mm	0°	+30°	0°

válcové výstupy, které přímo zapadnou to kostek, které mají ve středu každé z šesti stěn díru, která odpovídá délce a průměru válcových výstupu na čelistních. Kartézské souřadnice TCP bodu chapadla jsou vypsány v tabulce 3.2.



Obrázek 3.4: Chapadlo připevněné na robotu [27]

3.3 KUKA KR C4

Řídicí systém KR C4 (obrázek 3.5) od firmy KUKA integruje robotický kontrolér, PLC, motion kontrolér a safety. Všechny tyto systémy sdílí společnou datovou infrastrukturu a databázi. Kontrolér podporuje jazyky KRL, PLC a CNC. PLC má přístup k I/O robotu, díky kterým lze ovládat robotu. Pomocí funkčních bloků lze číst a zpracovávat proměnné. CNC nabízí programování a provoz robotů KUKA prostřednictvím G-kódu. KR C4 integruje safety kontrolér do řídicího systému, bez využití dalšího hardwaru. Bezpečnostní funkce jsou realizovány prostřednictvím rozhraní Ethernetu. Systém KR C4 podporuje 25 jazyků. Kontrolér KR C4 je nabízen v pěti variantách, KR C4 compact je využit v buňce. Ostatní kontroléry (KR C4 smallsize-2, KR C4, KR C4 midsize, KR C4 extended) se oproti compact liší ve velikosti, krytí vůči prahu a vodě, podporovaných os, hmotnosti a napájení. Parametry kontroléru KR C4 compact jsou vypsány v tabulce 3.3. [29]

3.4 KUKA smartPAD

KUKA smartPAD byl vyvinut pro snadné zvládnutí komplexních úkolů, který slouží k ovládání robotů KUKA s řízením KR C4 nebo Sunrise Cabiets KUKA. SmartPAD obsahuje dotykový kapacitní



Obrázek 3.5: KR C4 Compact [27]

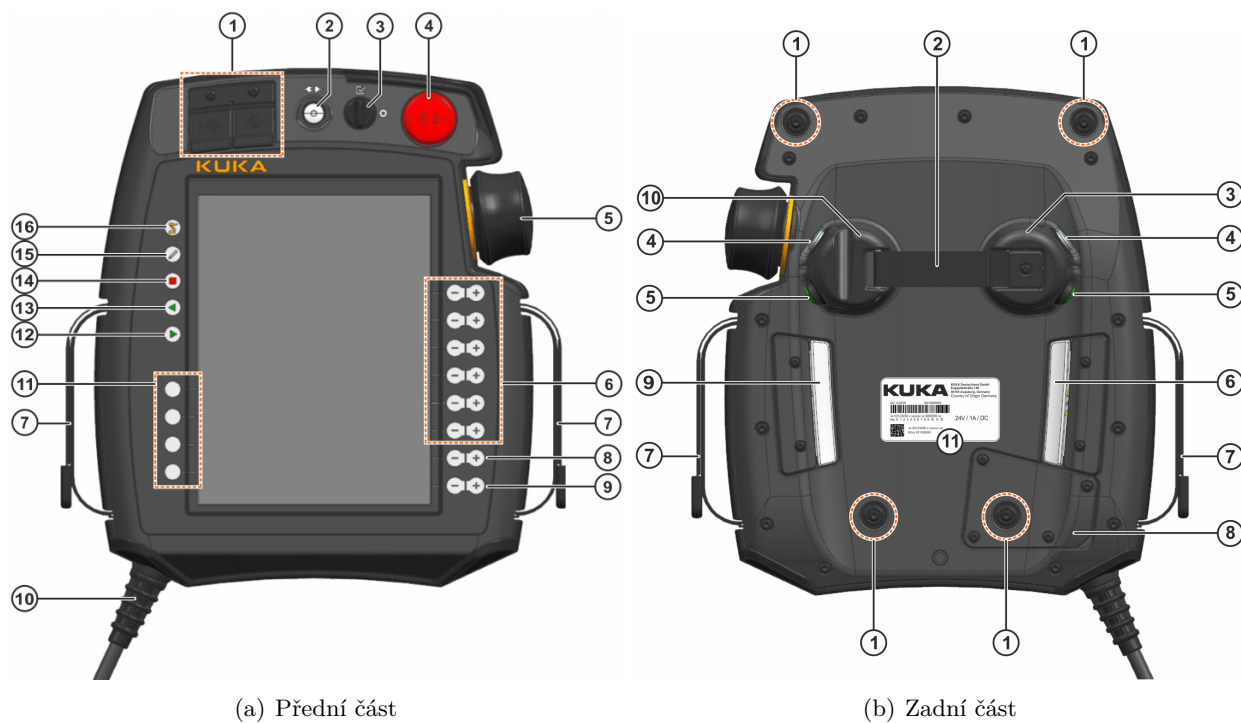
Tabulka 3.3: Parametry KR C4 compact [27]

KR C4 compact	
Rozměry (VxŠxH)	271x483x460 mm
Procesor	Technologie MultiCore
Pevný disk	SSD
Rozhraní	USB3.0, GbE, DVI-I
Maximální počet os	6+2
Síťová frekvence	50/60 Hz \pm 1
Jmenovité připojené napětí	AC 200 V až 230 V
Druh ochrany	IP20
Okolní teplota	+5°C až +45°C
Hmotnost	33 kg

8,4 palců velký display s třídou oblasti IP 54, který je odolný proti poškrábání. SmartPAD nabízí 6D myš s intuitivním postupem a možnosti změny orientace robotu ve třech nebo ve všech šesti stupních volnosti. Na přední straně smartPADu se nacházejí dva USB konektory, které umožňují načtení uživatelských programů. [30]

KUKA smartPAD lze vidět na obrázku 3.6. ① Dvě rozhraní USB2.0 s kytem. ② Tlačítko pro odpojení smartPAD. ③ Přepínač druhů provozu. ④ Tlačítko nouzového zastavení. ⑤ 6D myš. ⑥ Pohybové klávesy k ručnímu ovládaní robotu. ⑦ Ruční poutka se suchým zipem. ⑧ Tlačítko k nastavení override programu. ⑨ Tlačítko k nastavení ručního override. ⑩ Připojovací kabel. ⑪ Stavová tlačítka. ⑫ Tlačítko Start, kterým se spouští program. ⑬ Tlačítko Start-Zpět. ⑭ Tlačítko STOP. ⑮ Klávesa klávesnice. ⑯ Klávesa hlavního menu. [27]

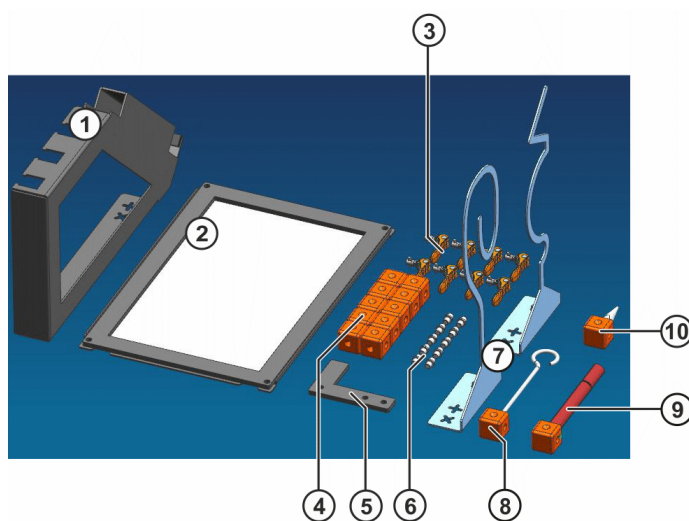
Zadní strana KUKA smartPAD je vidět na obrázku 3.6. ① Možnost připevnění nosného popruhu. ② Poutko. ③ Držení smartPADu pravou rukou. ④ Potvrzovací spínač. ⑤ Tlačítko Start. ⑥ Potvrzovací spínač. ⑦ Ruční poutka se suchým zipem. ⑧ Kryt připojovacího kabelu. ⑨ Potvrzovací spínač. ⑩ Držení smartPADu levou rukou. ⑪ Typový štítek. [27]



Obrázek 3.6: KUKA smartPAD

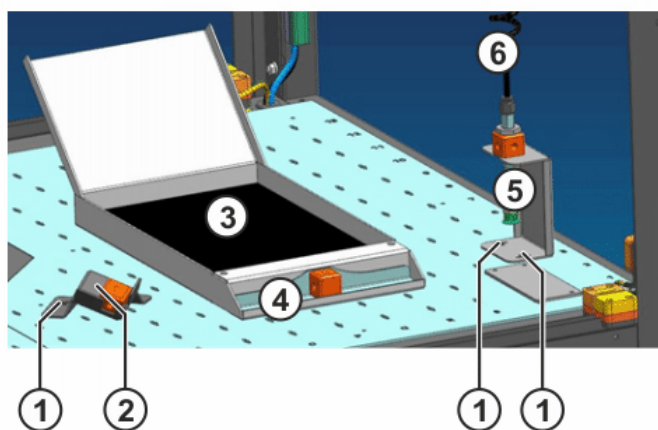
3.5 Aplikační prvky

Buňka obsahuje základní nástroje pro trénování (obrázek 3.7). ① Zásobník na kostky. ② Držák na list papíru. ③ Excentrické páčky. ④ Kostky 25x25 mm. ⑤ Úhelník. ⑥ Regálový čep. ⑦ Horký drát. ⑧ Kostka s háčkem. ⑨ Kostka s barevnou tužkou. ⑩ Kostka s měřícím hrotem.



Obrázek 3.7: Základní aplikační prvky [27]

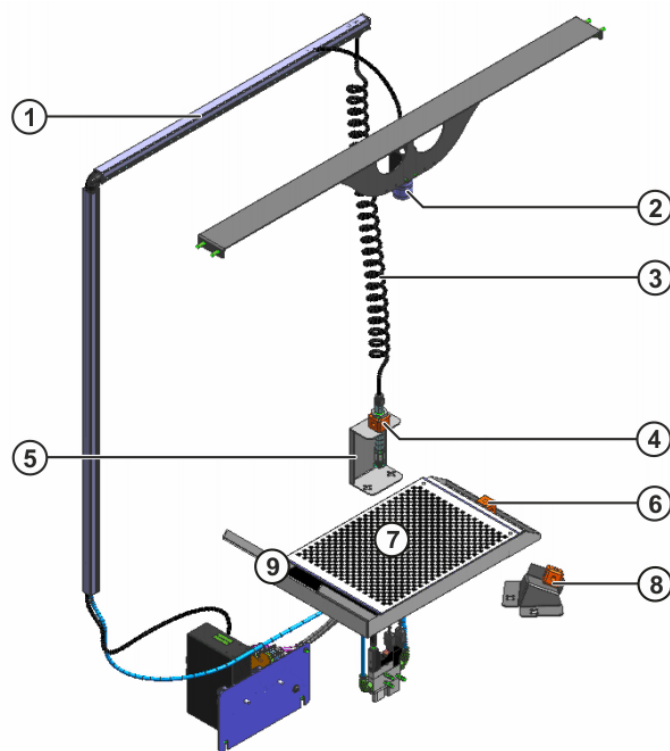
Rozšířené aplikační prvky pro Vision (obrázek 3.8). ① Excentrická páka. ② Centrovací stanice. ③ Skluzný žlab s regálovým čepem a kalibrační deskou. ④ Pravitko s kostkou. ⑤ Kostka se sacím zařízením a držákem. ⑥ Spirálová hadice. Robotické pracoviště obsahuje systém Vision, který je složen z (obrázek 3.9). ① Kabelový kanál. ② Kamera s objektivem. ③ Spirálová hadice. ④ Kostka se sacím zařízením. ⑤ Držák pro sací zařízení. ⑥ Pravitko s kostkou. ⑦ Kalibrační deska. ⑧ Centrovací stanice. ⑨ Skluzavka. [27]



Obrázek 3.8: Vision aplikační prvky [27]

3.6 Uživatelské prostředí

Po zapnutí buňky hlavní vypínačem (viz. obrázek 3.2) a načtení systému se zobrazí uživatelské prostředí. To je zobrazené na obrázku 3.10. ① Stavový řádek (popsán o odstavec níž). ② Zobrazuje počet zpráv a jejich typ. ③ Okno zpráv, zobrazuje se pouze poslední zpráva, dotykem na zprávu se otevře. Poslední zprávu lze potvrdit tlačítkem "OK", nebo potvrdit všechny zprávy "Confirm All". ④ Zobrazení aktuálního zvoleného souřadnicového systému pro jogging pomocí 6D myši, kliknutím na tuto ikonu lze měnit souřadný systém. ⑤ Indikace zarovnání 6D myši a nastavení úhlu, ze kterého je rameno programováno, cílem je, aby např. při pohybu 6D myši od sebe rameno jelo od uživatele. ⑥ Zobrazení aktuálního zvoleného souřadnicového systému pro jogging pomocí tlačítek, kliknutím na tuto ikonu lze měnit souřadný systém. ⑦ Jogovací klávesy, které se mění podle navoleného jogování. Při jogování pomocí kloubů se zobrazí A1 až A6. Při jogování pomocí souřadného systému se zobrazí X, Y, Z, A, B, C ⑧ Změna rychlosti robotu během provádění programu. ⑨ Změna rychlosti robotu při jogování (rychlost nesmí překročit 250 mm/s). ⑩ Panel tlačítek, které vždy odkazují na aktivní okno. Tlačítkem "Edit" lze upravovat, kopírovat, mazat a provádět mnoho dalších možností se soubory. ⑪ Ikona WorkVisual, kliknutím na tuto ikonu se otevře okno správy projektu. ⑫ Zobrazení systémového času s datem. ⑬ Ikona skupiny uživatelů, která zobrazuje bílé segmenty



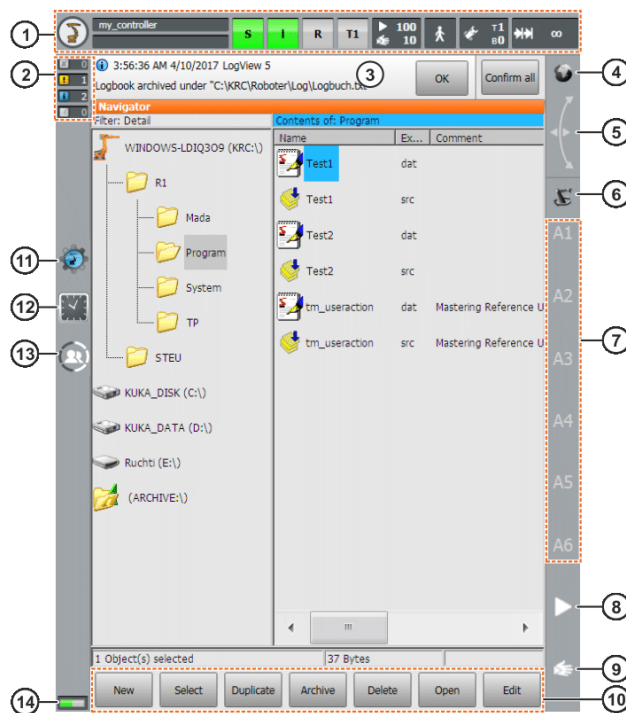
Obrázek 3.9: Vision systém [27]

v kruhu. Počet svítících segmentů označuje, jaká skupina uživatelů je označena. ⑭ Ikona zobrazuje znamení o životě.

3.6.1 Stavový řádek

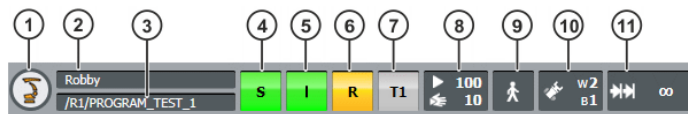
Stavový řádek označuje stav určitých nastavení robotu. Stavový řádek je popsán na obrázku 3.11.

① Tlačítko hlavní nabídky, odkazující na položky v uživatelském rozhraní (skupiny uživatelů, konfigurace, diagnostika a další). ② Název robotického kontroléru. ③ Název navoleného programu. ④ Indikátor stavu interpreter. ⑤ Indikátor stavů pohonů, který má tři stavy. Zelený se symbolem I značí zapnutí pohonů. Šedý se symbolem I potvrzuje, že již byly potvrzovacího spínače sepnuty, ale momentálně jsou rozepnuty. Šedý se symbolem O potvrzuje, že potvrzovací spínače nebyly sepnuty a nebo existují aktivní zprávy zabraňující pohyb robot. Dále při stisku ikony zobrazuje sepnutí ochranu obsluhy. ⑥ Indikátor stavu programu a možnost resetovat nebo zrušit právě navolený program. Ikona má pět stavů. Šedá ikona značí, že nebyl zvolen žádný program. Žlutá ikona oznamuje, že ukazatel je na prvním řádku navoleného programu. Zelená ikona značí vykonávání zvoleného programu. Červená ikona oznamuje zastavení navoleného programu. Černá ikona potvrzuje dokončení programu. ⑦ Zobrazení aktuálně zvoleného režimu. ⑧ Nastavení rychlosti pohybů při provádění



Obrázek 3.10: Uživatelské prostředí [27]

programu a jogování. ⑨ Indikátor stavu běhu programu. ⑩ Zobrazení navoleného nástroje a základny. ⑪ Stavové zobrazení.



Obrázek 3.11: Stavový řádek [27]

3.7 PLC

V rozvodné části robotické buňky je na DIN liště Umístěno PLC SIMATIC ET 200SP (obrázek 3.12) s CPU 1512SP-1 PN s verzí firmwaru v2,8. Tato procesorová jednotka obsahuje pracovní paměť 200 kB pro program a 1 MB pro data. Tato jednotka obsahuje Profinet interface s třemi porty. PLC podporuje vytvoření OPC-UA serverů, který může například zajistit komunikaci mezi jednotkou a Process Simulate. OPC-UA server běží na straně PLC a výměna dat probíhá pomocí názvů proměnných. IP adresa PLC je 172.31.1.147.100.



Obrázek 3.12: SIMATIC ET200 s CPU 1512SP-1

3.8 HMI panel

V přední části buňky (viz. obrázek 3.2) se nachází HMI panel od firmy Siemens a to SIMATIC HMI KTP 900 Basic (obrázek 3.13). Panel má devíti palcovou uhlopříčku displeje a osm tlačítek. Barevný rozsah displeje je 65536 barev a rozlišení displeje je 800 x 480 pixelů. Panel je napájen 24V DC a má Profinet rozhraní. Konfigurovat panel lze pomocí softwarů WinCC Basic V13 nebo STEP 7 Basic V13. IP adresa HMI je 172.31.1.147.150.



Obrázek 3.13: SIMATIC HMI KTP 900 Basic

3.9 Skupiny uživatelů

Právě zvolenou skupinu uživatelů lze poznat dle počtu bílých segmentů v uživatelském prostředí. Po zapnutí buňky je automaticky přihlášen operátor. Pokud je režim přepnut na AUT nebo AUT EXT, řídicí jednotka robotu se z bezpečnostních důvodů automaticky přepne na operátora. Při neprovádění akcí delších jak pět minut je aktuální uživatel odhlášen a přepnut na operátora (dobu lze změnit). Heslo pro všechny skupiny uživatelů je v základu "kuka". Skupiny uživatelů s popisem a počtem segmentů zobrazených jsou vypsány v tabulce 3.4.[27]

Tabulka 3.4: Skupiny uživatelů [27]

Skupiny uživatelů		
Skupina	číslo	popis
Operator	1	Omezené práva, uživatel nemůže provádět funkce, které mění systém.
User	2	Tento uživatel může provádět funkce, které jsou potřeba pro normální provoz robotu.
Expert	3	Uživatel může provádět funkce, které vyžadují odborné znalosti.
Safety recovery technician	4	Uživatel může provádět funkce potřebné pro údržbu systému.
Safety maintenance technician	5	Tato skupina může provádět funkce pro spuštění systému, včetně bezpečnostního systému.
Administrator	6	Skupina může provádět skoro veškeré funkce, až na bezpečnostní.

3.10 Volba pracovního režimu

Volba pracovního režimu se provádí pomocí přepínače na smartPADU (obrázek 3.6). Přepínač může obsahovat zámek a nebo ne. Režimy přepínat pouze pokud robot neprovádí žádný program. Jednotlivé pracovní režimy jsou vypsány v tabulce 3.5. [27]

Tabulka 3.5: Pracovní režimy [27]

Pracovní režimy		
Pracovní režim	Použití	Rychlosti
T1	Pro zkušební provoz, programování a učení	Ověření programu, maximální rychlost 250 mm/s, režim jogování má stejnou maximální rychlosti
T2	Testovací režim	Ověření programu, programovatelné rychlosti, režim jogování není dostupný
AUT	Pro průmyslové roboty bez nadřazeného řízení	Mód programu, programovatelné rychlosti, režim jogování není dostupný
AUT EXT	Pro průmyslové roboty s nadřazením řízením např. PLC	Mód programu, programovatelné rychlosti, režim jogování není dostupný

Kapitola 4

Digitální dvojče robotické buňky

Digitální dvojče pracoviště bylo vytvořeno v prostředí Tecnomatix Process Simulate (dále jen PS) a to ve verzi 15.1. Při vytváření projektu je potřeba dbát na cestu, kde byl projekt vytvořen. Projekt pro tuto práci je uložen v cestě `C:\PLM\Process_Simulate_data\Projects\DiplomaThesisVicherek`, pro otevření tohoto projektu je potřeba dodržet cestu. Pro správné fungování projektu je potřeba v záložce options a disconnected nastavit tuto cestu do client system root.

4.1 Import modelu

Model celé robotické buňky byl poskytnut firmou KUKA. Model musí být exportován do formátu JT, aby jej bylo možno importovat do PS. Import modelu probíhá přes funkci CAD translator, kterou lze doinstalovat do prostředí. Při importu je potřeba zvolit jestli komponenta se bude chovat jako part a nebo resource.

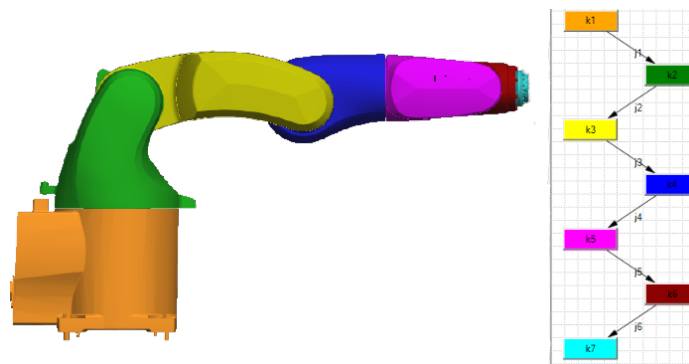
Part slouží jako materiál, ze kterého je zhotoven celý produkt, tvoří se u něj tok materiálu. Resource slouží jako nástroj k vytvoření hotového produktu, např. dopravníky, robotická ramena, a další. Po zvolení se vytvoří v cestě uložení projektu soubor komponenty s příponou COJT.

Model buňky obsahoval veškeré komponenty v jednom resource objektu, proto jednotlivé části byly přesunuty a vytvořeny pro tyto objekty samostatné resource komponenty a byly uloženy do cesty projektu. Vytvářená resource komponenta se dále parametrizuje dle jejího použití, např. kontejner, robot, úchyt, chapadlo, dopravník, člověk, pistole, zařízení a další. Většina objektů (nástroje, kostky, držáky, konstrukce buňky atd.) v projektu je definována jako device. Robot a chapadlo jsou definovány dle svého využití jako robot a chapadlo. Part byl využit jeden, ale pouze pro testování automatického vytváření trajektorií, které nebylo využito v úloze.

Projekt je vytvářen tak, aby bylo možné, co nejjednodušší využít dané modely pro rozdílné úlohy.

4.2 Robotické rameno KR 3 R540

Model robotického ramene byl již součástí buňky, ale tento model nebyl využit a byl smazán. Na stránkách výrobce KUKA lze stáhnout rameno s vytvořenou kinematikou a parametry rychlostí a zrychlení jednotlivých kloubů. Model tohoto ramene je již v .COJT formátu a stačí jej pouze nadefinovat a poté vložit do prostředí, použitím funkcí define component a insert. Stažený soubor obsahuje data, která se nahrají do ramene v PS při použití RCS modulu, tento modul nebyl využit. RCS modul zajistí přesnější simulaci. Kinematiku ramene lze vidět na obrázku 4.1, barevné obdélníky obsahují mechanické části jednotlivých os ramene (pojmenovány k1-k7), mezi obdélníky jsou šipky, které reprezentují jednotlivé klouby s jejich maximálními úhly natočení, rychlostmi a zrychleními.



Obrázek 4.1: Kinematika ramene v PS

V nástroji pose editor je vytvořena home pozice robotu, do které se robot vrací na koncích operace. Natočení jednotlivých kloubů odpovídá home pozici u reálného ramene, která je nadefinována v kontroléru robotu. Natočení jednotlivých kloubů v home pozici lze vidět na obrázku 4.2.

Edit Pose - kr3_r540				
Joins tree	Steering/Poses	Value	Lower Limit	Upper Limit
kr3_r540	HOME			
j1		0,00	-170,00	170,00
j2		-110,00	-170,00	50,00
j3		107,00	-110,00	155,00
j4		0,00	-175,00	175,00
j5		3,00	-120,00	120,00
j6		-1,00	-350,00	350,00

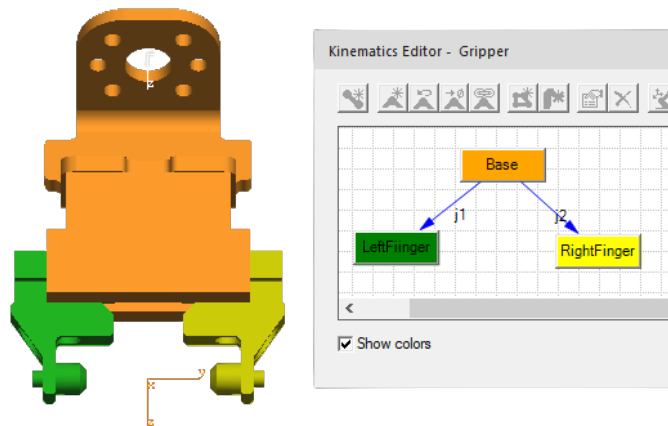
Pose name: HOME

Reset OK Cancel

Obrázek 4.2: Home pozice

4.3 Chapadlo

Model chapadla byl nadefinován v prostředí jako gripper. Pro správnou funkčnost chapadla, je potřeba vytvořit kinematiku pohyblivých částí, v tomto případě kleštin. Chapadlo s jeho kinematikou lze vidět na obrázku 4.3. Pravá a levá čelist chapadla se jednotlivě pohybují po lineární ose ($j1$ a $j2$). Vytvořená kinematika nabízí možnost vytvářet pozice chapadla, které by mělo mít minimálně dvě pozice. Pro toto chapadlo jsou vytvořeny dvě pozice a to pozice rozevření a sevření.

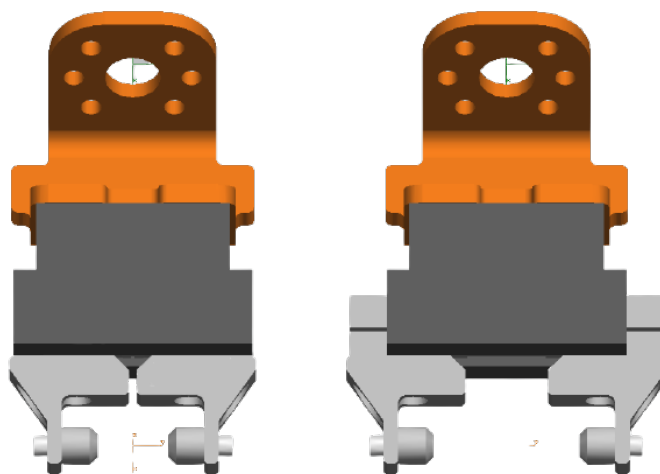


Obrázek 4.3: Chapadlo a jeho kinematika v PS

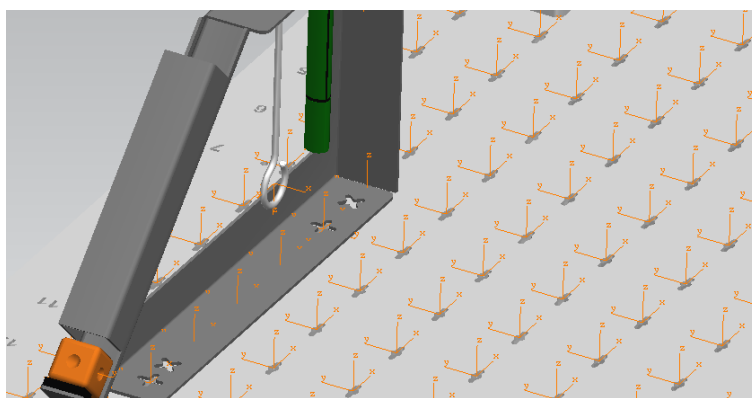
Pro nadefinování chapadla je zapotřebí vytvořit Base a TCP frame. Base frame slouží k připojení k robotickému rameni (obrázek 4.3 bílý osový kříž). TCP frame (Tool Center Point) pomocí kterého se poté rameno orientuje v souřadném systému (obrázek 4.3 oranžový osový kříž). TPC bod je natočen podle parametrů z tabulky 3.2. Při definování chapadla se využívá funkce v PS, ve které se nastavují parametry base frame, TCP frame a části chapadla, které nezajišťují a zajišťují dotyk se sevřeným objektem. Části chapadla, které se při sevření dotýkají, s přemísťovaným objektem zajistí, že při vyzvednutí objektu zůstane objekt v kleštinách. Pozice otevření a zavření lze vidět na obrázku 4.4.

4.4 Montážní deska

Pro jednoduchou změnu rozložení nástrojů v buňce a tvorbu robotických operací jsou vytvořeny body (v PS se tyto body nazývají frames). Body jsou vytvořeny na nástrojích a částech buňky (zásobník na kostky, kostka s hrotem, montážní deska a konstrukce na světlo). Body jsou rozmístěny různě na místa dle potřeby daného prvku, ale jsou vždy osově stejně natočeny. Montážní deska má nad otvory umístěny body, které jsou pojmenovány jak v reálné buňce (od A až U a číselně od 1 do 13), pomocí těchto bodů lze jednoduše upravovat rozmístění nástrojů pomocí nástrojů po přesouvání, které PS nabízí, viz obrázek 4.5.



Obrázek 4.4: Otevřená a zavřená pozice chapadla



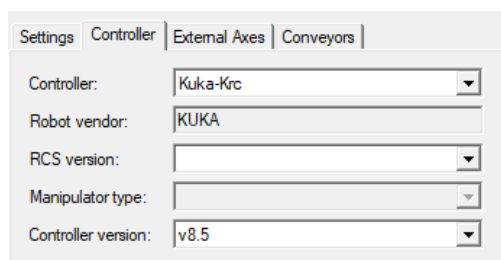
Obrázek 4.5: Rozmístění bodů na montážní desce

4.5 Nastavení OLP v Process Simulate

Pro tvorbu kódu je potřeba mít nainstalovaný balíček OLP KUKA pro Process Simulate 15, který je ke stažení na Siemens PLM Download Center. Pro úspěšné vygenerování funkčního kódu je potřeba nastavit odpovídající verzi kontroléru robotu vůči reálnému provedení, definovat báze a nástroje.

4.5.1 Nastavení verze kontroléru robotu

Verze kontroléru robotu v digitálním prostředí musí odpovídat tomu reálnému. Process Simulate nabízí tři druhy kontroléru pro robotická ramena KUKA a to KUKA-krc, KUKA-Volvo a KUKA-BMW. Kontrolér pro tuto aplikaci byl zvolen KUKA-krc a verzi 8.5, která odpovídá reálné verzi kontroléru u buňky (viz. obrázek 4.6). Verzi krc kontrolérů nabízí Process Simulate od verze 4.x do 8.7.



Obrázek 4.6: Nastavení kontroléru

4.5.2 Definování nástrojů aází v digitálním dvojčeti

Definování nástrojů aází se provádí v nastavení robotu v záložce robot setup. Souřadnicový systém báze definuje polohu vzhledem ke světovému souřadnicovému systému. Nástrojový bod definuje souřadnicový bod nástroje. Nadefinované báze jsou vypsány v tabulce 4.1 a nástroje v tabulce 4.2. Definována báze byla pouze jedna a to souřadný počátek celého prostředí PS. Do této pozice byla přesunuta pata robotu s celou buňkou.

Tabulka 4.1: Nadefinované báze v Process Simulate

Parametry vytvořenýchází							
číslo báze	název	X	Y	Z	A	B	C
0	Báze robotu	0,00	0,00	0,00	0,00	0,00	0,00

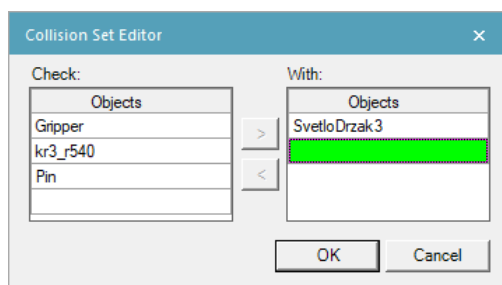
Tabulka 4.2: Nadefinované nástroje v Process Simulate

Parametry vytvořených nástrojů							
číslo nástroje.	název	X	Y	Z	A	B	C
9	TPSTIP	152,29	0,00	-59,80	0,00	30,00	0,00
10	TPSGripper	109,29	0,00	-34,80	0,00	30,00	0,00

4.6 Detekce kolizí

PS nabízí detekci kolizí, navolí se objekty, které mají hlídat kolizi mezi sebou. Při vzniku umí PS zastavit simulace, nebo pokračovat v simulaci a při kolizi se přehraje zvuková stopa. Objektům, které jsou v kolizi, lze přiřadit barvu pro zjednodušenou vizualizaci kolize. Robot, chapadlo a nástroj byly kontrolovány vůči modelu světlometu, původně byl v detekci i zásobník na kostky, ale při uložení nástroji PS detekoval kolizi. Navolení kolize lze vidět na obrázku 4.7.

Detekce kolize byla vytvořena, ale nebyla využita. Při zapnutí detekce nebylo možné efektivně využívat nástroje pro vytváření operací, protože byla celá simulace zasekaná. Problém vznikl nejspíš



Obrázek 4.7: Rozmístění bodů na montážní desce

s nízkým HW vybavením počítače, na kterém byl projekt tvořen, a vysokým rozlišením modelu světlometu.

Kapitola 5

Návrh úlohy pro ověření offline programování

Tato kapitola popisuje návrh úlohy pro offline programování robotu, tvorbu konstrukce pro model světloometu a úpravu digitálního modelu světloometu, aby jej šlo vložit do prostředí PS.

5.1 Návrh úlohy

Robotické rameno bude pracovat se světlometem. Operace, které lze se světlometem provádět, jsou omezeny díky jeho velikosti, hmotnosti a vybavenosti buňky. Základem budou operace, které budou využívat kostku s měřícím hrotem (viz. obrázek 3.7.) k objíždění hran světloometu.

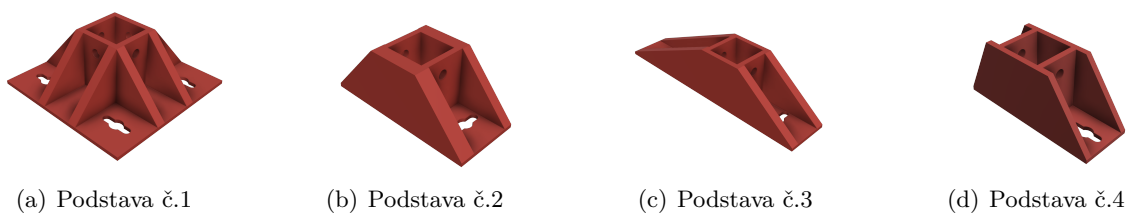
Úloha bude vytvořena a realizována pomocí nástroje Tecnomatix Process Simulate, kde budou veškeré robotické operace vytvořeny, parametrizovány a exportovány do reálného kontroléru robotu. Úloha bude poté testována na reálném prostředí, a jestli budou potřeba úpravy, budou prováděny opět v prostředí Tecnomatixu.

5.2 Model světloometu

Světlomet byl z fyzické podoby přenesen do digitální s použitím 3D skeneru. Výsledný digitální model světloometu je ve formátu STL a byl dodán firmou, která prováděla skenování. Model musel být upraven, protože prostředí Process Simulate a Fusion 360 jej nedokázal naimportovat díky vysoké kvalitě. Původní STL model se skládal z přibližně 35 miliónů trojúhelníků, po snížení kvality má model okolo jednoho miliónu trojúhelníků. Ke snížení kvality byl využit program Meshmixer. Pro upevnění světloometu na montážní desku robotické buňky byla navržena konstrukce.

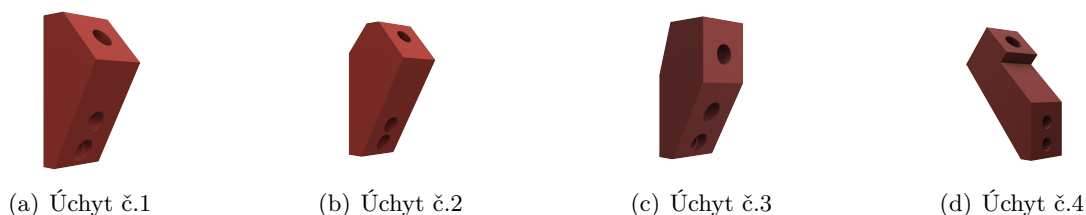
5.3 Konstrukce

Konstrukce pro úchyt světlometu se skládá ze čtyř podpěr. Každá podpora se skládá z podstavy, hliníkového profilu a úchytu, který uchytí světlomet k profilu. Podstavy a úchyty jsou vytvořeny pomocí technologie 3D tisku, která nabízí jednoduché otestování a následné úpravy, jakmile něco neseďí. Modely byly vytvořeny v prostředí Fusion360. Montážní deska robotické buňky má jednotlivé montážní otvory 50 mm vzdálené od sebe a vytváření tak sítě, do které lze cokoli upevnit. Světlo met má zvolené úchyty různě vzdálené, proto je každá část jedinečně navržena pro dané místo uchycení. Profily byly nařezány na délky 150 mm, 155 mm, 185 mm a 230 mm. Některé podstavy prošly několika verzemi, jelikož se během návrhu opomnělo na natočení montážních úchytů a umístění excentrických pák. Podstavy lze vidět na obrázku 5.1 a úchyty na obrázku 5.2.



Obrázek 5.1: Díly podstav

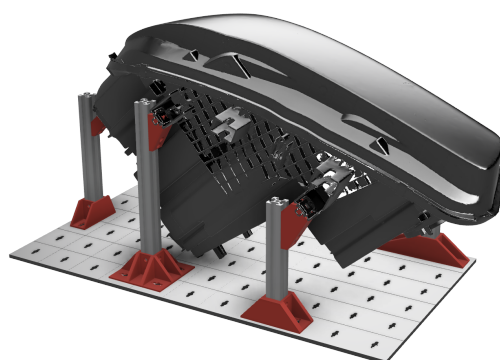
Každá podstava a úchyt jsou k profilu připevněny pomocí M5 šroubů a matic, spojení mezi úchytem a světlometem je pomocí M8 šroubů. Celá sestava je umístěna přímo před robotickým ramenem. Podstava a úchyt č.1 jsou spojeny profilem o délce 230 mm, podstava a úchyt č.2 profilem délky 150 mm, podstava a úchyt č.3 profilem délky 155 mm a poslední úchyt a podstava č.4 profilem délky 185 mm. Světlo met je natočen přední částí směrem k robotickému rameni. Na obrázku 5.2 je pohled na celou sestavu světlometu a konstrukce. Konstrukce se světlometem byly importovány do Process Simulate a vloženy na pozici.



Obrázek 5.2: Díly úchytů



(a) Přední pohled na světlomet



(b) Zadní pohled na světlomet

Obrázek 5.3: Sestava světlometu

Kapitola 6

Návrh a realizace programu

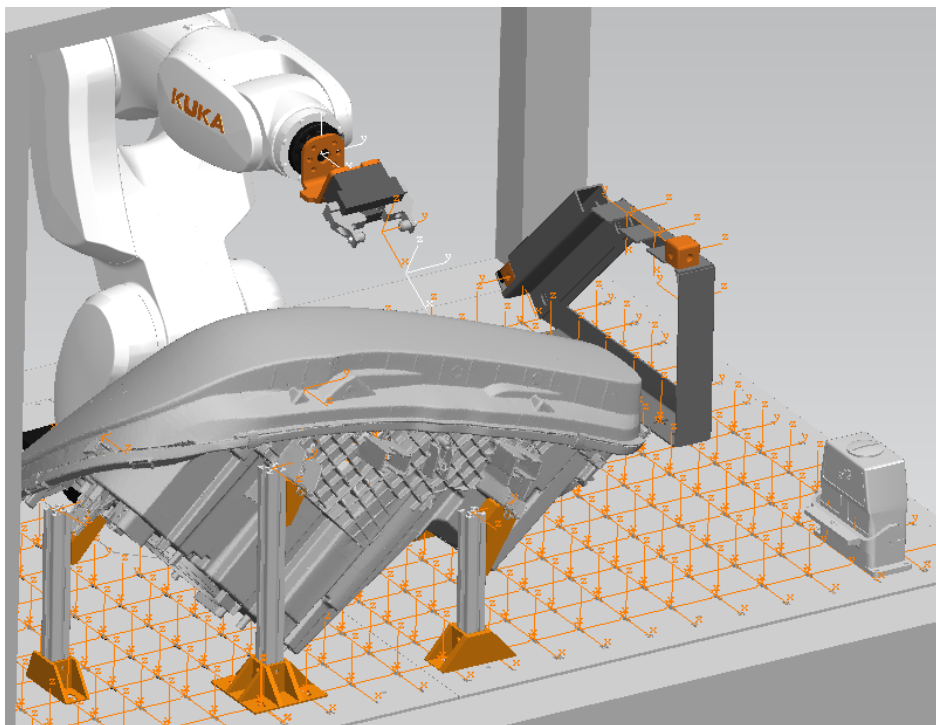
V této kapitole je popsán postup, jak byly robotické operace v programu Process Simulate vytvářeny. Popsány jsou také parametry, které byly využity a jak byly parametrizovány. Výsledkem této kapitoly je postup vytváření funkčních programů, které stačí nahrát do reálného zařízení.

6.1 Návrh programu

Návrh úlohy byl jednoduchý seznam úkonů, které má robotické rameno vykonat, aby byla ověřena možnost offline programování s využitím PS. Z HOME pozice robot najede pro kostku s hrotem, uchytí ji a přesune se nad zvolenou hranu světlometu. Objede hranu světlometu a vrátí nástroj zpět na pozici, odkud jej odebral a vrátí se zpět do home pozice. Tento koncept návrhu úlohy byl dodržen, ale přibyla jedna operace navíc, která najíždí nad profily a sjíždí hrotem na úroveň montážního otvoru. Proto byl poté navržen program, který bude volat jednotlivé operace, podle toho jakou si operátor navolí.

6.2 Rozložení pracoviště

Světlomet je umístěn přímo před robotem, aby měl robot možnost pracovat s celým prostorem. Rozmístění jednotlivých podpor lze vidět na obrázku 5.3. Podstava č.1 je fixovaná k montážní desce na pozicích H1, H2, G1 a G2, podstava č.2 na pozicích K1 a L1, podstava č.3 na pozicích L6 a N3, poslední podstava č.4 na pozicích D3 a D4. Zásobník na kostky s držáky na nástroje je umístěn na pozicích U12 a U8. Nástroj je umístěn na první pozici (strana, kde se nenachází zásobník na kostky). Aby bylo možné tuto úlohu opět bez problému spustit, je potřeba dodržet tohoto rozmístění. Rozložení lze vidět na obrázku 6.1.



Obrázek 6.1: Rozložení pracoviště

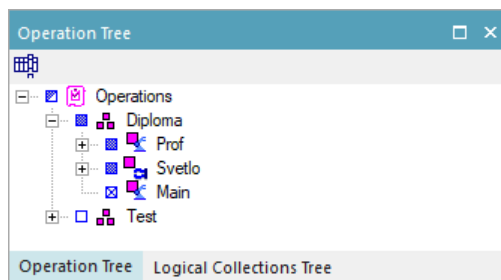
6.3 Tvorba operací

Při vytváření robotické operace v PS, je zapotřebí mít nadefinovaný robot s chapadlem, nebo jiným nástrojem v projektu. Během vytváření operace nabízí PS čtyři druhy operací.

1. Svařovací operace - využívá se pro tvorbu svařovacích operací, během vytváření se volí pistole a pozice svarů, které se mají simulovat
2. Pick and place operace - využívá se pro tvorbu operací, které manipulují s objekty, během vytváření se volí chapadlo, robot a pozice nabrání a uvolnění objektu
3. Operace s kontinuálními prvky - aplikuje se pro laserové svařování, nanášení lepidla a nástřik
4. Obecná robotická operace - vytvoří se prázdná operace bez bodů, navolí se nástroj, robot, body se dotvářejí poté pomocí nástrojů

Pick and place a obecná robotická operace byla zvolena při tvorbě programů. Takto vytvořené operace se zobrazí v operation tree a je zapotřebí pro tyto operace vytvořit složku (tzv. Compound operation) a přesunout je do ní. Když operace nebudou ve složce, PS vyhodí při přepnutí modu ze standardního do simulačního chybu a je potřeba celé prostředí zapnout znovu. V projektu jsou v hlavní složce operations vytvořené dvě složky. První složka s názvem "Diploma" obsahuje zhotovené

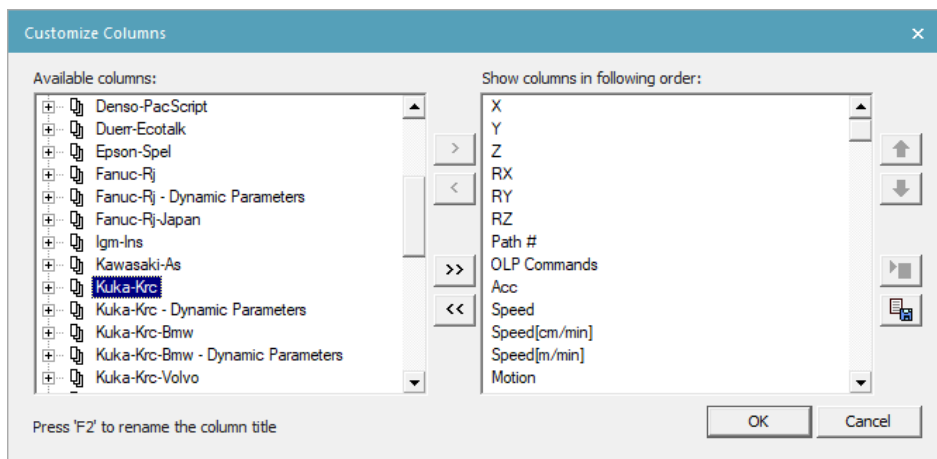
operace. Druhá složka Test obsahuje testovací programy, které nebyly využity ve finálním programu. Strom operací (operation tree) lze se složkami vidět na obrázku 6.2.



Obrázek 6.2: Operation tree

6.4 Path editor

Path editor se využívá k vytváření jednotlivých bodů robotických operací. Editor v základu nabízí pro daný bod pouze název, souřadnicový systém (X, Y, Z, RX, RY, RZ) a nabídku OLP příkazů. Přidání rozšíření KUKA Krc k programování KUKA robotů se provádí přes úpravu sloupců (funkce customize columns v PS). Do path editoru byly přidány sloupce z rozšíření KUKA-Krc. Na obrázku 6.3 lze vidět přidání KUKA-Krc. V levém okně lze vidět všechny dostupné nainstalované OLP rozšíření a v pravém okně jsou parametry.



Obrázek 6.3: Přidání KUKA-Krc

6.5 Vytváření a úprava bodů

Vytvářet body (tzv. Via) v operacích lze několika způsoby.

1. Přidání aktuální pozice - tímto způsobem se vytvoří na poslední pozici v operaci bod, který odpovídá aktuální pozici TCP bodu robotu.
2. Přidání lokace výběrem - místo, kde se má nacházet bod, se volí kliknutím myši, bod se vytvoří s natočením, které odpovídá globálnímu kartézskému systému, toto nemusí být úplně vhodné a je dobré si před vytvořit bod s natočením, který odpovídá TCP bodu
3. Přidání bodu před - přidá bod před vybraný bod v path editoru
4. Přidání bodu za - přidá bod za vybraný bod v path editoru

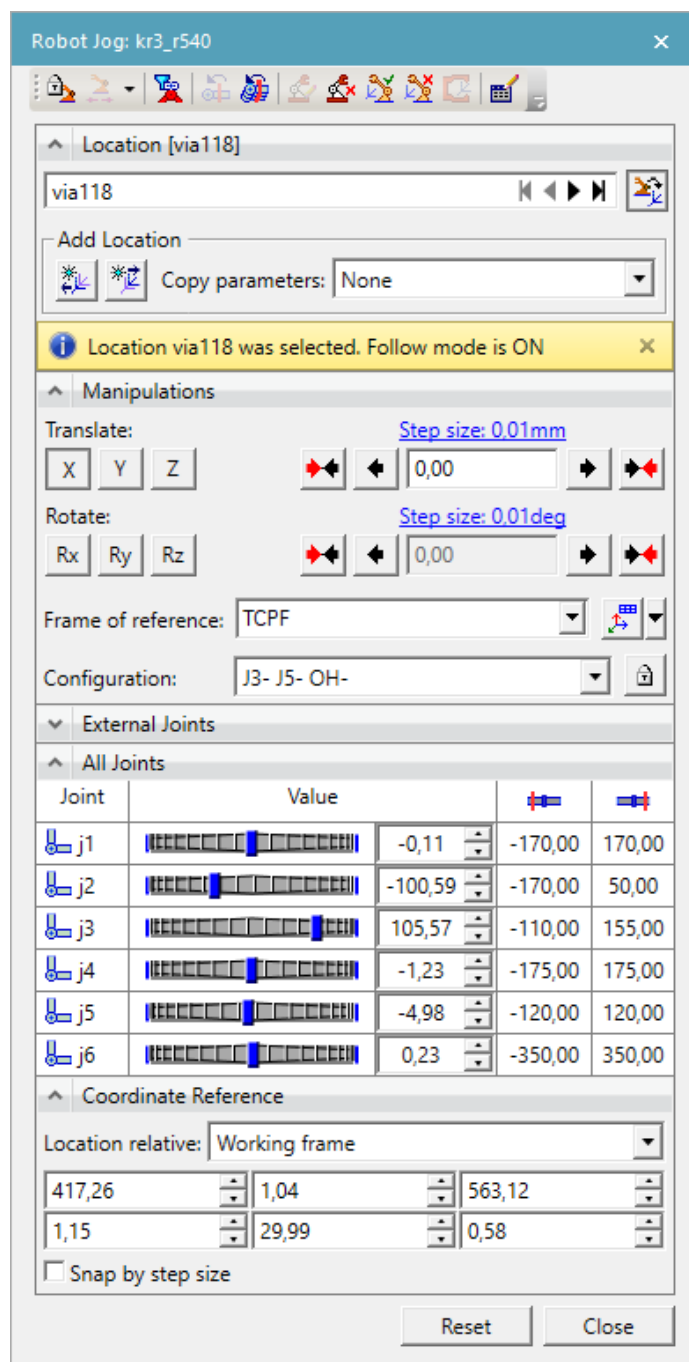
Při tvorbě pozic byly využity všechny čtyři nástroje dle potřeby. Manipulovat s těmito vytvořenými body lze přes klasické manipulační nástroje, přepisováním pozic X, Y, Z a natočení RX, RY, RZ v path editoru a pomocí nástroje manipulate location (obrázek 6.4). V horní části lze přepínat jednotlivé lokace, které je zapotřebí upravit. Manipulovat lokace lze ve všech osách, dle nastaveného kroku (tento krok je dobré upravit na nižší, v základu je nastavený na 100 mm). Dále lze měnit bod (Frame of reference), od kterého jsou přepočítány posuny. Velkou výhodou tohoto nástroje je zobrazení natočení robotu v prostředí, lze pak jednoduše vidět, v jaké pozici se robot nachází, jestli nedochází ke kolizi a lze s ním manipulovat přes TCP bod.

6.6 Parametrizace bodů

Parametry, které nabízí balík v PS pro offline programování robotů, odpovídají názvům parametrům v reálném kontroléru. Vypsání parametrů v path editoru a jejich hodnoty lze vidět na obrázku 6.5.

Základní parametry, které byly využity při parametrizování bodů:

- OLP Commands - Nástroj k tvoření programu v jednotlivých bodech, volání signálů, nástroj je více rozepsán v kapitole 6.9.
- Acc - zrychlení TCP do daného bodu
- Speed - rychlost TCP, u PTP pohybů byla rychlost volena 5% a u LIN, CIRC a SPL pohybů 0,05 a 0,02 m/s.
- Motion - druh pohybu, voleny byly PTP, LIN, CIRC a SPLINE. Principy pohybů jsou vypsány v kapitole 6.7
- Config - nastavení parametrů set, turn a učení pozic, podrobnější popis se nachází v kapitole 6.8
- Base nr - číslo báze, čísla využitých bází jsou vypsána v tabulce 4.1



Obrázek 6.4: Nástroj manipulate location

- Tool nr - číslo nástroje, pomocí kterého se má orientovat, využitě nástroje jsou vypsaný v tabulce 4.2
- Circ type - Definování orientace u CIRC pohybů, typy jsou popsány v kapitole 6.7.3

- Zone - Nastavení aproximace v daném bodě, nabízen je pouze parametr C_DIS, který aproximuje, jakmile vzdálenost do koncového bodu klesne pod nastavenou hodnotu
- Ori Type - Nastavení přechodu orientace TCP mezi počátečním a koncovým bodem, nastavuje se u LIN a CIRC pohybů, detailní popis se nachází v kapitole 6.7.2

Paths & Locations	X	Y	Z	RX	RY	RZ	Path #	OLP Co...	Acc	Speed	Speed[c...	Speed[m/min]	Motion	Config	Base Nr	Tool Nr	Circ Type	Zone	Ori Type
via118	417.26	1.04	563.12	1.15	29.99	0.58			75 %	5 %	5 %	5 %	PTP	S 6 T 27	0 - Worl...	10 (Fr) - ...		C_DIS 5 mm	
via21	146.29	472.56	390.00	-5.92	90.00	84.09			75 %	5 %	5 %	5 %	PTP	S 2 T 11	0 - Worl...	10 (Fr) - ...		C_DIS 5 mm	
via25	146.29	472.56	364.50	-5.92	90.00	84.09		# Destination	75 %	0.05 m/s	300 cm/...	3 m/min	LIN	S 2 T 11	0 - Worl...	10 (Fr) - ...	#PATH	FINE	VAR
via	146.29	472.56	366.50	-5.92	90.00	84.09			75 %	0.05 m/s	300 cm/...	3 m/min	LIN	S 2 T 11	0 - Worl...	9 (Fr) - T...	#PATH	C_DIS 5 mm	VAR
via20	211.14	300.51	378.00	0.00	90.00	0.00							CIRC	S 2 T 35					
via3	211.15	163.01	400.00	0.00	90.00	0.00			75 %	0.05 m/s	300 cm/...	3 m/min	CIRC	S 2 T 35	0 - Worl...	9 (Fr) - T...	#PATH	C_DIS 10 mm	VAR
via122	211.15	148.00	395.00	0.00	90.00	0.00							CIRC	S 2 T 35					
via119	211.15	143.01	380.00	0.00	90.00	0.00			75 %	0.05 m/s	300 cm/...	3 m/min	CIRC	S 2 T 35	0 - Worl...	9 (Fr) - T...	#PATH	C_DIS 10 mm	VAR
via6	211.15	143.01	330.00	0.00	90.00	0.00			75 %	0.02 m/s	120 cm/...	1.2 m/min	LIN	S 2 T 35	0 - Worl...	9 (Fr) - T...	#PATH	FINE	VAR
via130	211.15	143.01	380.00	0.00	90.00	0.00			75 %	0.02 m/s	120 cm/...	1.2 m/min	LIN	S 2 T 35	0 - Worl...	9 (Fr) - T...	#PATH	FINE	VAR

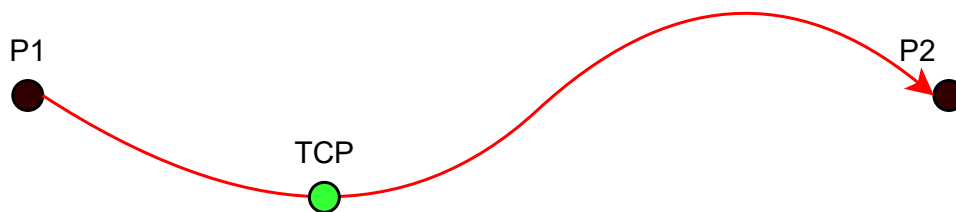
Obrázek 6.5: Path editor

6.7 Typy pohybů

Pohyby jsou voleny v záložce motion (obrázek 6.5). Pohyby, které byly využity v operacích, jsou popsány níže v kapitole.

6.7.1 PTP

Robot vede TCP nejrychlejší cestou ze startovního bodu (P1) do koncového bodu (P2). Nejrychlejší cesta obecně není nejkratší cestou a není tedy přímkou. Jelikož jsou osy robotu rotační, robot zakřivené cesty provádí rychleji než přímé cesty. Přesnou trajektorii nelze předvídat. Trajektorie je zobrazena na obrázku 6.6. [31]



Obrázek 6.6: PTP pohyb

6.7.2 LIN

Robot vede TCP po přímce mezi počátečním bodem P1 a koncovým P2. Trajektorie je zobrazena na obrázku 6.7. Orientace TCP se může lišit v počátečním a koncovém bodě pohybu. Existuje několik různých typů přechodu od počáteční ke konečné orientaci. [31]

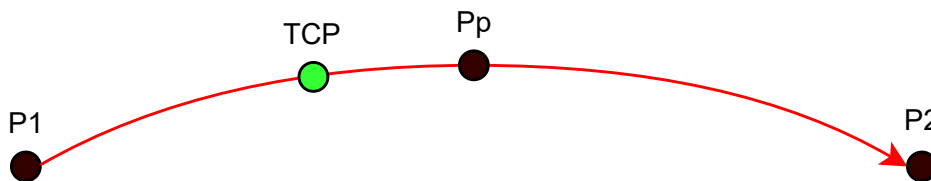
- Constant - Orientace TCP zůstává během pohybu konstantní, naprogramovaná orientace je ignorována pro koncový bod a orientace počátečního bodu je zachována.[31]
- Var - Orientace TCP se během pohybu neustále mění, pokud robot prochází singularitou, použijte místo toho Wrist PTP.[31]
- Wrist PTP - Orientace TCP se během pohybu neustále mění, ale ne rovnoměrně. PTP na zápěstí tedy není vhodné, pokud je třeba přesně dodržovat konkrétní orientaci, např. v případě laserového svařování.[31]



Obrázek 6.7: LIN pohyb

6.7.3 CIRC

Robot vede TCP definovanou rychlostí po kruhové cestě do koncového bodu. Kruhová cesta je definována počátečním (P1), pomocným (Pp) a koncovým bodem (P2). Trajektorie je zobrazena na obrázku 6.8. V PS se tento pohyb definuje prvním bodem, který je pomocný, druhý bod je koncový. PS automaticky při volbě pomocného bodu nabídne pouze volbu CIRC pohybu. Nastavené parametrů rychlostí a zrychlení u prvního bodu zruší a parametrizuje se pouze druhý bod.[31]

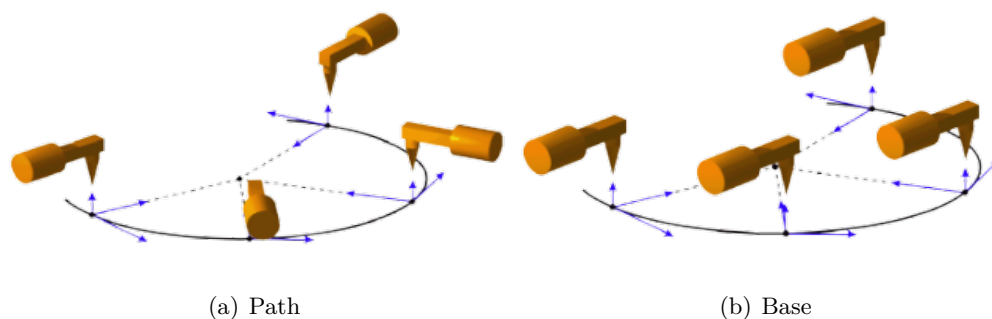


Obrázek 6.8: CIRC pohyb

U pohybů CIRC je také možné definovat (parametr Circ Type), zda má být ovládání orientace spojeno se základnou nebo s cestou.

- Path - Řízení orientace na trajektorii během CIRC pohybu [31]
- Base - Ovládání orientace související se základnou během CIRC pohybu [31]

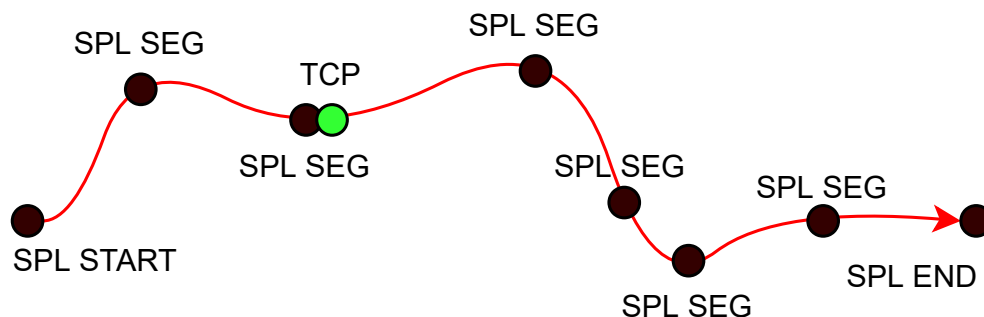
Vizuální zobrazení těchto dvou parametrů CIRC pohybu lze vidět na obrázku 6.9



Obrázek 6.9: Vizuální zobrazení parametru circ type

6.7.4 SPL

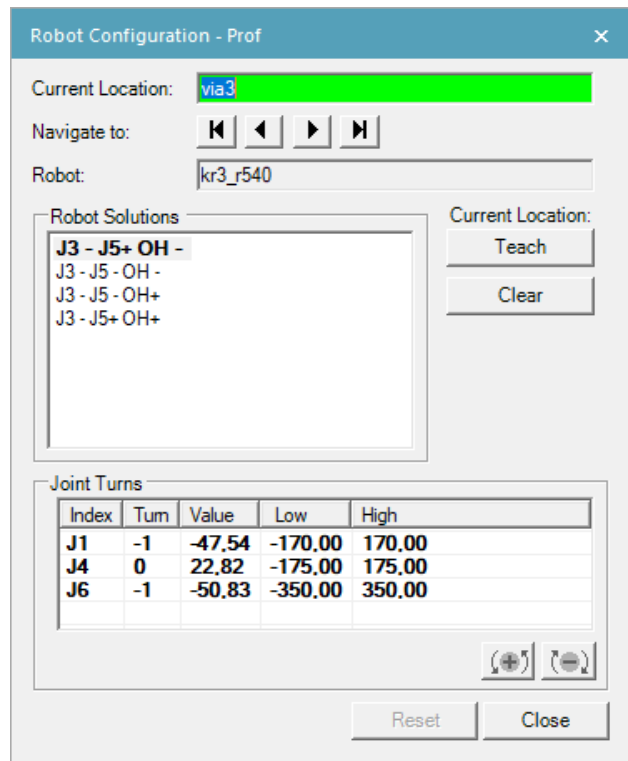
Spline je typ pohybu, který je vhodný zejména pro složité zakřivené cesty. Takové cesty lze také generovat kombinací pohybů LIN a CIRC, avšak SPL mají výhody. SPL se používá k seskupení několika pohybů jako celkového pohybu. SPL je naplánován a proveden řídicí jednotkou robotu jako jeden blok pohybu. Pohyby, které mohou být zahrnuty do spline bloku, se nazývají spline segmenty (SPL SEG). V PS se navolí první bod jako startovní (SPL START), body přes které má TCP procházet jsou navoleny jako spline segmenty (SPL SEG), poslední bod je navolen jako koncový (SPL END), zde blok končí. Na obrázku 6.10 lze vidět SPL pohyb s jeho body. [31]



Obrázek 6.10: SPL pohyb

6.8 Robot configuration

Sloupcem config (obrázek 6.5) se parametrizují jednotlivé body trajektorie. Na obrázku 6.11 se parametrizuje bod via3, do kterého se může rameno dostat čtyřmi způsoby natočení. Zvoleno bylo nejvhodnější natočení a tlačítkem "teach" je tento bod naučen a vygenerovány parametry status a turn. Každý bod může mít odlišný počet způsobů natočení, záleží na pozici bodu.



Obrázek 6.11: Robot configuration blok

6.8.1 Turn

Umožňuje pohybovat osami v úhlech větších než $+180^\circ$ nebo menších než -180° . Každá osa reprezentuje jeden bit, jakmile natočení osy $\geq 0^\circ$ bit je roven 0. Bit je roven 1, když je osa natočená $< 0^\circ$. [31]

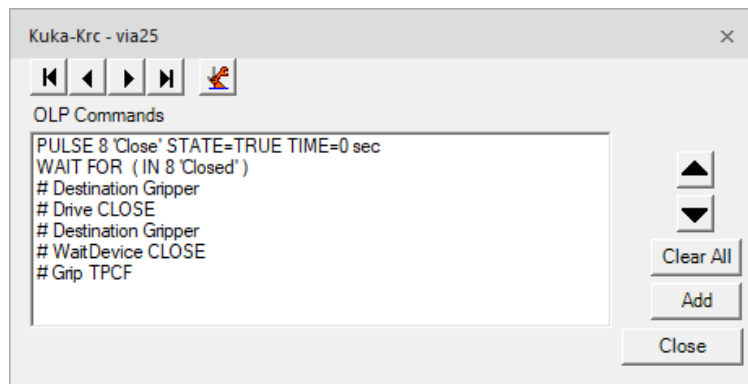
6.8.2 Status

Skládá se ze čtyř bitů, které spolu s pozicí TCP, definuje polohu osy robotu. Bit 0 určuje polohu průsečíku os A4, A5 a A6, průsečík vychází z podstavy robotu, jakmile je průsečík v kladné části osy A1 je bit roven 0, když do kladné části je roven 1. Bit 1 závisí na pozici osy A3, když je osa $\geq 0^\circ$ je vit roven 1, jinak 0. Bit 2 je závislý na naklonění osy A5 s odkazem na polohu A4. Jakmile je osa A5 nakloněna nahoru je bit roven 1, jinak je roven 0, toto platí i pro natočení osy pod úhlem 0° . Bit 3 není využit a je vždy roven 0. Bit 4 je pouze informativní a nemá žádný vliv na pozici. Bit určuje, jak byl daný bod naučen, jestli pomocí absolutně přesného robotu, nebo offline. Když byl bod naučen offline je bit roven 0, jinak 1. [31]

Parametry status a turn jsou v PS zobrazovány jako dekadické číslo. Body musejí být vždy naučeny, jinak nelze vyexportovat program z PS.

6.9 OLP Configuration

Nástroj, pomocí kterého lze posílat hodnoty na výstupy a číst vstupy, ovládat chapadlo, spínat pistole a psát komentáře. Obsahuje klasické programovací struktury jako je if, else, while, case. Obsahuje příkazy, které řídí určité části v PS, jako je ovládání chapadla, pistole, volání synchronizačních signálů, spojování materiálů do celku (např. simulace skládání nějakého produktu pomocí robotického ramene) a další. OLP configuration obsahuje předpřipravené příkazy pro KUKA kontroléry. Mezi ně patří řízení chapadla, vytváření komentářů, zápas a čtení signálů a detekci kolize.



Obrázek 6.12: Ukázka OLP příkazu

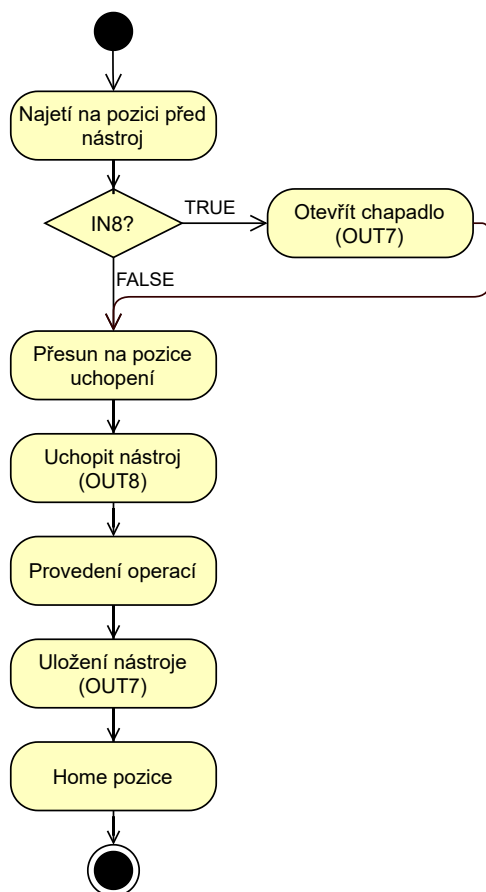
Na obrázku 6.12 lze vidět sekvenci příkazů, první dva jsou pro řídicí jednotku robotu. Příkazy se nachází v bodě, kde je zapotřebí uchopit pomocí chapadla nástroj. První příkaz posílá pulz na výstupní signál č.8, který sevře chapadlo. Druhý příkaz čeká na potvrzení správného sevření kleštin vstupním signálem č.8. Zbylé příkazy řídí chapadlo v PS. Do výsledného kódu nejsou exportovány, tato možnost byla zvolena v nastavení, jinak se příkazy exportují jako komentáře.

Příkaz vytvářející ovládání chapadla v PS pro jazyk krl, během kontroly programu v reálném kontroléru robotu nefungoval. Kontrolér se odkazoval na špatnou syntaxi, z tohoto důvodu bylo sevření a otevření chapadla řízeno přes výstupní signály.

6.10 Vytvořené trajektorie

S využitím PS byly vytvořeny dvě operace, které byly následně využity do hlavního programu. Veškeré body trajektorií, parametrizování, hlídání vstupů a výstupů bylo vytvářeno skrz PS. Obě operace začínají a končí v home pozici robotu. Dále obě operace používají hrot, který musejí prvně uchytit a na konci operace jej vrátit zpět. Před uchopením hrotu je zkontrolován vstup 8, který informuje o sevření kleštin. Když by byly kleštiny sevřeny, jsou pomocí vstupního 7 signálu otevřeny a pokračuje se k pozici uchopení. Když budou kleštiny otevřeny, pokračuje se k pozici uchopení. Kleštiny se sevrou pomocí signálu 8. Jakmile má robot v kleštinách uchopen hrot, pokračuje v operaci dále. Hrot je na konci operace umístěn zpět na pozici vyzvednutí a po otevření kleštin robot

najíždí do home pozice. Využité signály s popisy jsou vypsány v tabulce 5.1. Uchycení a odložení nástroje jsou popsány v aktivitním diagramu na obrázku 6.13.



Obrázek 6.13: Aktivitní diagram operací

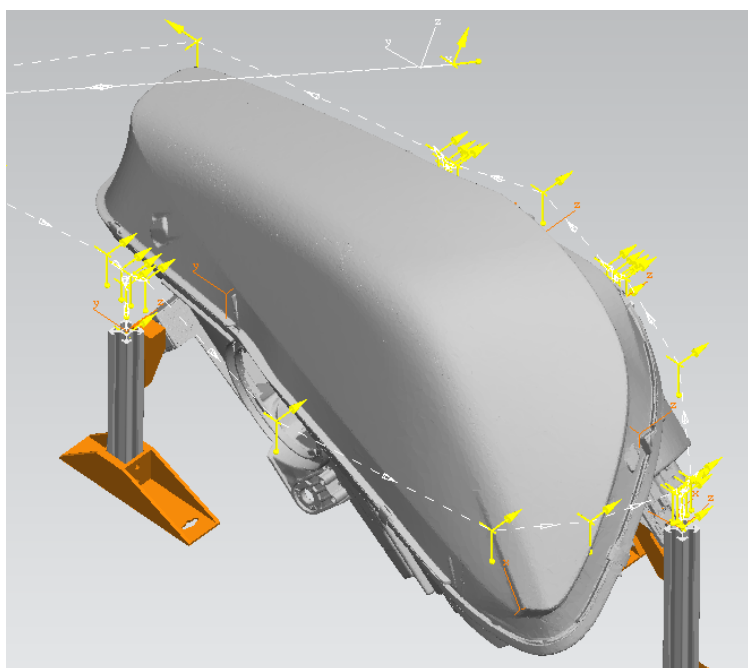
6.10.1 Operace profily

První operace s použitím hrotu najíždí nad kruhový otvor v profilu. Tato operace nebyla nijak plánována a vznikla díky testování přesnosti digitálního modelu a reality. Prvotní trajektorie obsahovala pouze profil s podstavou a úchytem č.3. Robot s nástrojem najel do určené výšky nad středový bod kruhového otvoru profilu a pomalu se posouval dolů s využitím LIN pohybu přesně na úroveň otvoru. Když TCP bod byl v místě otvoru, začal se vracet zpět na pozici odkud začal sjíždět dolů. Poté byly dodělané další profily s podstavami a úchyty č.4 , č.1 a č.2. Pohyb mezi jednotlivými profily probíhá pomocí CIRC a PTP pohybů. Najíždění nad otvor v určené výšce a následné sjíždět dolů po ose Z, bylo předěláno, aby přechod byl plynulejší. Proto byly přidány CIRC pohyby, které zajišťují plynulý pohyb na určenou výšku z předchozí pozice a následný pohyb po ose Z k bodu u otvoru. Na obrázku 6.14 lze vidět vytvořené body operace (žluté kříže). Bíle čerchované

Tabulka 6.1: Použité vstupní a výstupní signály

Seznam použitých vstupních a výstupních signálů				
Název proměnné	č.prom.	Typ	I/O	Popis
GripperAtOpen	7	Vstup	Boolean	Signalizace otevřeného chapadla
OpenGripper	7	Výstup	Boolean	Signál pro otevření chapadla
GripperAtClose	8	Vstup	Boolean	Signalizace zavřeného chapadla
CloseGripper	8	Výstup	Boolean	Signál pro zavření chapadla
StartSvetlo	200	Vstup	Boolean	Signál pro spouštění operace "Svetlo"
StartAlu	201	Vstup	Boolean	Signál pro spouštění operace "Profily"

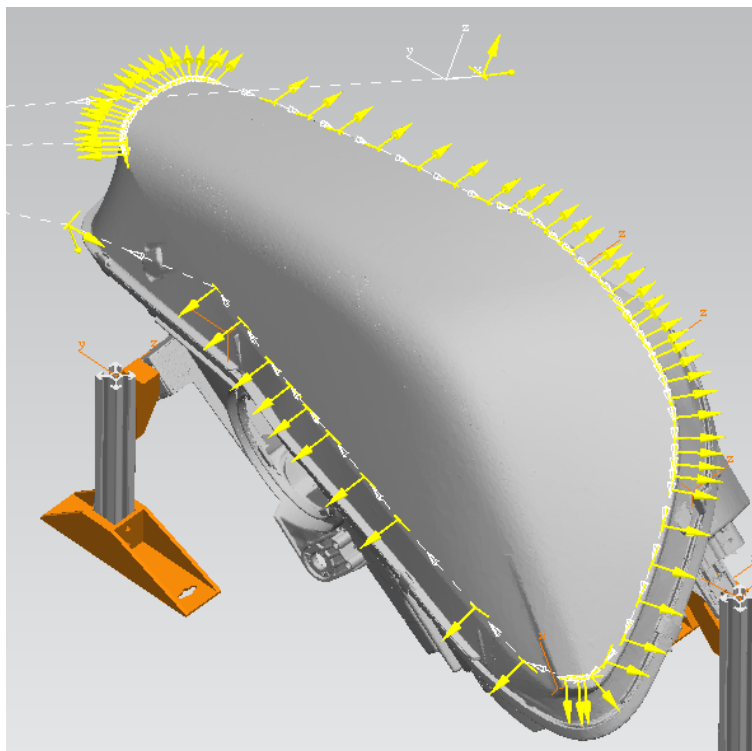
čáry nerepresentují, kudy bude veden TCP bod, ale kde začíná operace a kde končí. Nad profily lze vidět shluky bodů, které zajišťují plynulejší přechod.



Obrázek 6.14: Operace profily

6.10.2 Operace světlo

Druhá operace využívá hrot k objíždění světlometu. Jakmile má robot v chapadle hrot, přesune se k nejbližší straně světlometu a sjede nad hranu. TCP bod poté kopíruje hranu světlometu ve směru hodinových ručiček. Část operace, která objíždění světlometu, se skládá ze SPLINE pohybů. Na obrázku 6.15 lze vidět tuto operaci. Všechny body operací byly vytvořeny manuálně, nebyl využit žádný nástroj, který vytváří body pomocí křivek. Automatické generování i trajektorií nefungovalo, jelikož mode světlometu byl vytvořen s využitím 3D skeneru.



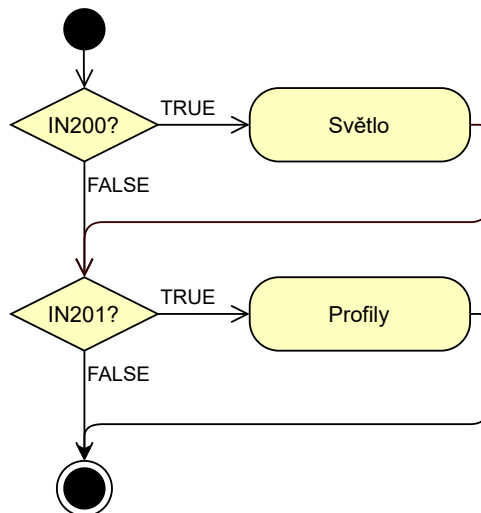
Obrázek 6.15: Operace světlo

6.11 Vytvoření programu robotu

Hlavní program robotu je vytvořen v PS jako obecná robotická operace, která neobsahuje žádné body. Název operace byl zvolen jako main. Logika řízení programu je vytvořena v OLP command bloku. Základem bloku je while loop, který vykonává v cyklu volání operací světlo a profily. Vždy je zavolána jedna z operací, dle vstupního signálu, který je sepnutý. Když je sepnutý vstupní signál 200, je zavolána operace světlo, při sepnutém signálu 201 je zavolána operace profily. Aktivitní diagram programu lze vidět na obrázku 6.16.

6.12 Exportování programu

Všechny tři operace byly v TPS vloženy do programu s názvem Diploma. Program se v PS vytváří pomocí nástroje create program inventory. Těchto programů může být více, ale navolen v PS může být pouze jeden. Operace se do programu vloží v path editoru. Program je vytvořen pro jednodušší export programu. Při exportu robotického programu, do kterého jsou vloženy všechny vytvořené operace, jsou exportovány všechny tyto operace. Tímto není třeba každou operaci samostatně exportovat.



Obrázek 6.16: Aktivitní diagram programu

Při exportu programů se vytvoří čtyři soubory. Dva soubory .dat a .scr jsou využity následně pro kontrolér robotu a jsou navoleny v reálném prostředí. Třetí soubor, který je exportován ve formátu .olp, obsahuje vytvořené nástroje, báze a pozice. Poslední soubor je textový soubor, který ukládá informace o provedeném převodu. V hlavičce je vypsán název šablony, podle které se vytvořily výsledné .dat a .src soubory. Pod hlavičkou jsou vypsány chyby a nenastavené parametry. Vypisuje nenaučené body, které nemají parametry turn a status. Při nezměněné rychlosti a zrychlení (ponechána základní) vypíše do logu, ve kterém body je nezměněná rychlost nebo zrychlení. Vypisování nezměněné rychlosti a zrychlení je velmi užitečné, protože PS nechává v bodu maximální rychlost a zrychlení při ponechání základního parametru.

Robotické operace lze i importovat do TPS. Vyexportovaná operace byla jednou naimportována z důvodu neuložení úprav a chybě v PS, která vypnula program.

Kapitola 7

Ověření funkčnosti

Vytvořené programy byly nahrávány a testovány v reálném zařízení, které odpovídá tomu digitálnímu. V této kapitole je popsán postup nahrání programu, provádění úprav digitálního modelu a testování.

7.1 Robot program viewer

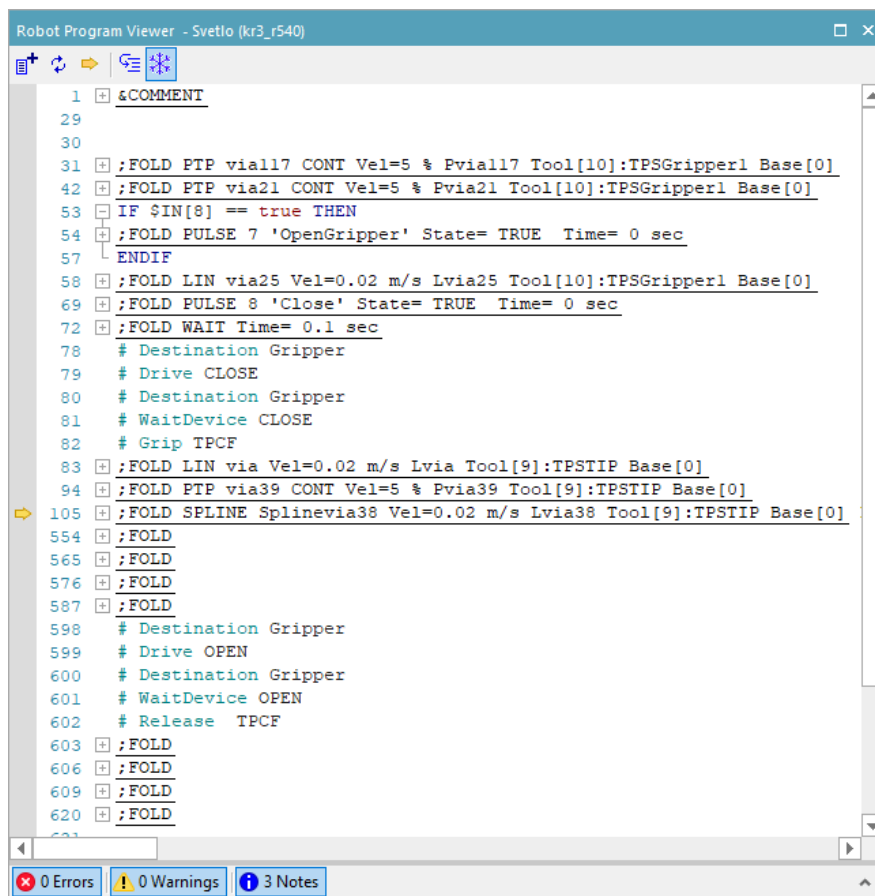
Nástroj robot program viewer slouží k zobrazení vytvořených programů a operací přímo v jazyce KRL v PS. Nástroj vypíše počet chyb, které se nacházejí v programu, ale neukáže, kde se chyba vyskytuje. Dále ukáže upozornění a poznámky k navolenému programu. Nástroj umožňuje sledovat kurzor, který ukazuje na místo, kde se program nachází. Dále lze přidávat breakpointy na zvolená místa v programu, v těchto místech se program zastaví. Nástroj slouží pouze k zobrazení a nelze v něm nijak editovat. Okno s navolenou operací světlo je na obrázku 7.1.

Nedostatky, které jsou zmíněny výše jsou v částečné míře odstraněny ve verzi 16.0.

7.2 Nahrání programu

Aby bylo možné otestovat vytvořené programy a trajektorie, musí se vytvořené trajektorie nahrát do reálného kontroléru robotu. Nahrání lze provést dvěma způsoby, pomocí flash disku, a nebo přes sdílenou složku. Obě varianty byly využity, ze začátku byl program nahráván přes flash disk. Když byl flashdisk připojen, zkopíroval se program a vložil do složky kontroléru robotu, kde byl navolen.

Přístup ukládání přes sdílenou složku je jednodušší a pohodlnější. Exportovat programy z PS lze rovnou do sdílené složky. K připojení je potřeba změnit IPv4 adresu ethernetového portu počítače. Tato adresa musí být 172.31.1.xxx, ale nesmí být .147 – Robot, .100 – PLC, .150 – HMI. K sdílené složce se lze připojit pomocí průzkumníku Windows a složky sítě. Zadáním adresy "\\172.31.1.147" vyskočí přihlašovací okno (uživatel: krcexpertprogrammer, heslo: kuka). Do složky "Sandbox" byly poté



Obrázek 7.1: Robot program viewer

exportovány všechny vytvořené programy. Nahrané programy ve sdílené složce bylo potřeba poté přemístit do složky robotu, aby mohly být navoleny a spuštěny.

7.3 Testování trajektorií na reálném zařízení

Prvotní testování se provádělo v režimu T1 (pracovní režimy jsou popsány v tabulce 3.5). Navolené operace ze začátku najížděly špatně na místo uchycení kostky s hrotem. Problém vznikl v nástrojích, reálné nástroje (chapadlo a hrot) mají v reálném prostředí jinde TCP bod, díky nepřesnostem. Z tohoto důvodu byly nástroje v reálném prostředí upraveny. Nástroje z PS a kontroléru jsou vypsány v tabulce 4.2.

Chapadlo bylo řízeno pomocí výstupních signálů. Blok na ovládání chapadla, který je v PS parametrizovatelný, při exportu nefungoval. Při parametrizaci se zadává číslo chapadla, to v reálném kontroléru je 1. Dalším parametrem je sevření nebo otevření chapadla, tato možnost se volí dle potřeby. Během exportu program nenahlásil žádnou chybu se špatně nadefinovaným ovládáním chapadla a program byl nahrán do reálného kontroléru, kde ale nešel navolit. Reálný kontrolér se

odkazoval na špatnou syntaxi v ovládání chapadla. Z tohoto důvodu bylo tedy zvoleno řízení přes výstupní signály. Na potřebný výstupní signál je vždy posílán pulz, který sevře nebo otevře kleštiny chapadla. Výstupní signály jsou vypsány v tabulce 6.1

Tabulka 7.1: Nástroje v PS a upraven v kontroléru robotu

Parametry nástrojů v PS							
číslo nástroje.	název	X	Y	Z	A	B	C
9	TPSTIP	152,29	0,00	-59,80	0,00	30,00	0,00
10	TPSGripper	109,29	0,00	-34,80	0,00	30,00	0,00
Parametry nástrojů v reálné buňce							
9	TPSTIP	151,50	0,3	-60,2	0,0	30,00	0,00
10	TPSGripper	113,8	0,5	-33,8	0,00	30,00	0,00

Operace profily, která najíždí na střed otvoru v profilech, sloužila k ujištění shody obou prostředí. První profil ukázal neshodu a bylo potřeba jemně posunout digitální model. K posunutí konstrukce se světlometem v digitálním prostředí byla vyčítaná aktuální pozice TCP v reálném prostředí. Pozice TCP ramene se přesunula na úroveň montážního otvoru v profilu a aktuální pozice bodu byla využita v digitálním prostředí k přesunu profilu s konstrukcí na tuto pozici. Najíždění na ostatní profily po posunutí již probíhalo bez větších problémů. Veškeré vytvořené body operace s jejich parametry fungovaly bez problémů.

Při prvotním najíždění na hranu světlometu docházelo k problému s výškou, jelikož reálný model byl posazený o trochu výš než ten digitální. Po odladění výšky úchytů u jednotlivých profilů vytvořený program fungoval dobře.

Během testování byly odladěny rychlosti v některých částech pohybů u obou trajektorií. Čas obou operací se oproti časům v PS lišil. Rozdíl v časech operací byl způsoben kontrolérem v PS, který je základní a nabízen v PS. Aby časy a pohyby roboty odpovídaly těm reálným je potřeba využít RCS modul, který je jako balíček doinstalován do PS. Modul dokáže lépe simulovat reálný kontrolér robotu a je vyvíjen přímo firmou KUKA. Druhá možnost, jak zlepšit digitální prostředí je využít KUKA.OfficeLite (SW je popsán v kapitole 2.1.1), tato možnost je nabízená od verze 16 (práce byla vytvořena v 15.1). PS nabízí připojení k tomuto virtuálnímu kontroléru, který také nahrazuje ten základní v PS. Při nastavení aproximace v bodech jej PS nevykonává a projíždí body normálně, ale v reálném řešení robot provádí nastavenou aproximaci v bodě.

Kapitola 8

Zhodnocení výsledků

Cílem diplomové práce bylo ověřit offline programování robotu s využitím principu digitálního dvojčete. Vytvořený finální program v programu Tecnomatix Process Simulate byl exportován do reálného zařízení, kde byl testován a byla ověřena jeho funkčnost.

V první části diplomové práce je teoreticky popsán princip offline programování. Celý princip offline programování přináší mnoho výhod, jako je možnost programování bez potřebného HW, celkové zrychlení doby programování, kontroly doby taktu, vizuální kontrolu, ověření dosahů a další. Dále je provedena analýza trhu se softwary pro offline programování. Celkem bylo nalezeno a zapsáno třináct softwarů. Všechny byly důkladně popsány, vypsány jejich výhody a procesy, které podporují. Většina těchto zanalyzovaných softwarů nabízí všechny procesy (svařování, nástřik, řezání a další) v jednom celku. Některé firmy mají řešení těchto procesů jako samostatné produkty (např. Delfoi), a nebo se zaměřují pouze na jednu oblast jako Verbotic Weld, který se zaměřuje pouze na svařování. Programy bývají také rozděleny na různé verze, které zpřístupňují další nástroje a umožňují vytváření k komplexnějším úloh. To vše za navýšení finální ceny programu. Dále firmy nabízejí rozsáhlé knihovny robotických ramen a dalších komponent potřebných k vytvoření úlohy.

Laboratorní pracoviště ready2_educate nabízí velkou škálu možností. Buňka obsahuje aplikační prvky, jako je horký drát, kostky, hrot, fixu, držák na papír, kameru. Všechny tyto prvky lze využít k vytvoření vlastní aplikace. Pracoviště obsahuje na předním čisti HMI panel, pomocí kterého lze ovládat vytvořený program. Pracoviště obsahuje rameno KUKA KR 3 R540 s kontrolérem KR C4, do kterého je připojen smartPAD.

Postup tvorby digitálního dvojčete se skládal z několika důležitých kroků, aby bylo možné exportovat funkční kód. V první části je popsán postup importu modelů do prostředí Tecnomatix Process Simulate. Déle byla vytvořena kinematika u ramene a chapadla. Rameno se skládá z šesti rotačních os a chapadlo ze dvou lineárních. V dalším kroku byl nastaven kontrolér KUKA-krc verze 8.5. Vytvořené nástroje a báze ve virtuálním kontroléru, které byly poté exportovány do reálného zařízení. Báze byla vytvořena jedna a to v patě robotu. Nástroje byly dva, jelikož v operacích se

pracovalo s rozdílnými nástroji. Další funkce která byla nastavena je detekce kolizí, která nebyla využita kvůli nízkému HW výkonu počítače.

Návrh úlohy se věnuje vytváření konstrukce pro světlomet, který byl do digitální podoby přenesen pomocí 3D skeneru. Digitální model světlometu musel být upraven, protože Process Simulate jej nedokázal naimportovat díky jeho vysoké kvalitě. Po upravě modelu byla vytvořena konstrukce, která se skládá ze čtyř podstav, každá podstava má jeden hliníkový profil a dva díly vytvořené pomocí 3D tisku. Celá konstrukce je navržena tak, aby byla možnost ji přesouvat po montážní desce v buňce dle potřeby.

Výsledkem offline programování v Tecnomatix Process Simulate jsou dvě funkční operace a jeden program, který tyto operace volá, dle vstupních signálů z HMI panelu. Celá kapitola se věnuje vytváření jednotlivých operací a nastavování typu pohybů, programování v bodech, úpravě rychlostí a zrychlení, aproximaci a dalším. Program byl napsán v OLP command bloku, který je velmi nepraktický. Výsledný program se skládá z while smyčky, která se vykonává pořád dokola, dokud se nezavolá vstupní signál IN200 nebo IN201, každý signál je přiřazen k jedné operaci, která je poté spuštěna. Syntaxe příkazů se dá nalézt pouze v dokumentaci, která je uložená v instalační cestě OLP rozšíření. Napsaný kód není nijak kontrolován PS, pouze jej lze debugovat vizuálně spouštěním simulací nebo pomocí nástroje program viewer.

Závěr práce se věnuje popisu testování vytvořeného programu a operací. Digitální model musel být trochu upraven, aby odpovídal realitě. Dále byly upraveny nástroje v reálném prostředí. Poté exportované programy odpovídaly těm v digitálním dvojčeti. Vytváření a úprava operací v Process Simulate je velmi jednoduché a intuitivní a dokáže ušetřit hodně času oproti klasickému programování. Programování programu pomocí OLP command bloku už nebylo tak pohodlné, jelikož je to univerzální nástroj pro programování. Nezobrazuje chyby, je potřeba si pomáhat nástrojem program viewer, který zobrazí běh programu.

Literatura

1. MITSU, S.; BOUZAKIS, K.-D.; MANSOUR, G.; SAGRIS, D.; MALIARIS, G. Off-line programming of an industrial robot for manufacturing. *The International Journal of Advanced Manufacturing Technology*. 2005, roč. 26, č. 3, s. 262–267. ISSN 0268-3768. Dostupné z DOI: 10.1007/s00170-003-1728-5.
2. PAN, Zengxi; POLDEN, Joseph; LARKIN, Nathan; DUIN, Stephen van; NORRISH, John. Automated Offline Programming for Robotic Welding System with High Degree of Freedoms. *Advances in Computer, Communication, Control and Automation*. 2012, s. 685–692. ISBN 978-3-642-25540-3. Dostupné z DOI: 10.1007/978-3-642-25541-0_86.
3. *KUKA.Sim*. KUKA AG, 2020. Dostupné také z: https://www.kuka.com/cs-cz/produkty,-slu%C5%BEby/robotick%C3%A9-syst%C3%A9my/software/pl%C3%A1nov%C3%A1n%C3%AD-projektov%C3%A1n%C3%AD-servis-bezpe%C4%8Dnost/kuka_sim.
4. *KUKA.OfficeLite*. KUKA AG, 2020. Dostupné také z: https://www.kuka.com/cs-cz/produkty,-slu%C5%BEby/robotick%C3%A9-syst%C3%A9my/software/pl%C3%A1nov%C3%A1n%C3%AD-projektov%C3%A1n%C3%AD-servis-bezpe%C4%8Dnost/kuka_officelite.
5. *Tecnomatix*. Německo: Siemens, [b.r.]. Dostupné také z: <https://www.plm.automation.siemens.com/global/en/products/tecnomatix/>.
6. *Robotics and automation Simulation*. Německo: Siemens, [b.r.]. Dostupné také z: <https://www.plm.automation.siemens.com/global/en/products/manufacturing-planning/robotics-automation-simulation.html>.
7. *Tecnomatix Process Simulate brožura*. Německo: Siemens Industry Software, 2011. Dostupné také z: <https://www.axiomtech.cz/25343-tecnomatix-process-simulate-robotics>.
8. *Robot programming and simulation software*. Octopuz, [b.r.]. Dostupné také z: <https://octopuz.com/>.
9. *Visual Componnets*. Visual Componnets, [b.r.]. Dostupné také z: <https://www.visualcomponents.com/products/visual-components/>.
10. *Teaching Robots with Visual Componnets*. Visual Components, 2017. Dostupné také z: <https://www.visualcomponents.com/insights/articles/teaching-robots-visual-components/>.

11. *About Delfoi Oy.* Delfoi, [b.r.]. Dostupné také z: <https://www.delfoi.com/company/company/>.
12. *Delfoi Robotics.* Delfoi, [b.r.]. Dostupné také z: <https://www.delfoi.com/delfoi-robotics/>.
13. *Delfoi Arc Software.* Delfoi, [b.r.]. Dostupné také z: <https://www.delfoi.com/products/delfoi-robotics/delfoi-arc/>.
14. *Delfoi Auto ARC.* Delfoi, [b.r.]. Dostupné také z: <https://www.delfoi.com/products/delfoi-robotics/delfoi-auto-arc/>.
15. *Delfoi CUT.* Delfoi, [b.r.]. Dostupné také z: <https://www.delfoi.com/products/delfoi-robotics/delfoi-cut/>.
16. *Delfoi CUT.* Delfoi, [b.r.]. Dostupné také z: <https://www.delfoi.com/delfoi-robotics/delfoi-paint/>.
17. *Offline Programming with FASTSUITE.* FASTSUITE, [b.r.]. Dostupné také z: https://www.fastsuite.com/en_EN/products/offline-programming.html.
18. *CIROS Studio for 3D Factory Simulation -.* Dortmund: VEROSIM Solution, [b.r.]. Dostupné také z: <https://www.verosim-solutions.com/en/industry/ciros-studio/>.
19. *Robot Simulation and Programing.* RobotDK, [b.r.]. Dostupné také z: <https://robodk.com/>.
20. *Robotmaster CAD/CAM for robots.* [B.r.]. Dostupné také z: <https://www.robotmaster.com/en>.
21. *Robot Programming Software - fast,easy automatic.* Convergent Information Technologies GmbH – Schulstrasse 2, [b.r.]. Dostupné také z: <http://convergent-it.com/>.
22. *Automappps - TBM Automation AG Vison & Robotics.* TBM Automation AG, [b.r.]. Dostupné také z: <https://www.robots.ch/en/products/cit/automappps.html>.
23. *FAMOS robotic.* Carat robotic innovation GmbH, [b.r.]. Dostupné také z: <http://famos-robotic.de/index.php?id=homepos%5C&L=1>.
24. *Verbotics Weld - Verbotics.* North Wollongong: Verbotics, 2020. Dostupné také z: <https://verbotics.com/>.
25. *SprutCAM Robot.* 2019. Dostupné také z: <https://sprutcam.com/cs/sprutcam-robot/>.
26. *KUKA Education / KUKA AG.* KUKA AG, [b.r.]. Dostupné také z: <https://www.kuka.com/cs-cz/produkty,-slu%5C%C5%5C%BEby/robotick%5C%C3%5C%A9-syst%5C%C3%5C%A9my/kuka-education>.
27. *Application Module ready2educate Provozní návod.* Německo: KUKA, 2020.
28. *KR 3 AGILUS / KUKA AG.* KUKA, [b.r.]. Dostupné také z: <https://www.kuka.com/cs-cz/produkty,-slu%5C%C5%BEby/robotick%5C%A9-syst%5C%A9my/pr%5C%AFmyslov%5C%A9-roboty/kr-3-agilus>.

29. *KUKA KR C4 / KUKA AG*. KUKA, [b.r.]. Dostupné také z: <https://www.kuka.com/cs-cz/products/robotics-systems/robot-controllers/kr-c4>.
30. *KUKA smartPAD / KUKA AG*. KUKA, [b.r.]. Dostupné také z: <https://www.kuka.com/cs-cz/produkty,-slu%C5%BEby/robotick%C3%A9-syst%C3%A9my/%C5%99%C3%ADdic%C3%AD-syst%C3%A9my-robot%C5%AF/smartpad>.
31. *KUKA KSS 8 5 Manual for integrators en*. Německo, 2017.

Příloha A

Soubory

- UkázkaProgramu.MP4 - Sestříhané video s reálným a digitálním provedením operací
- DiplomaThesisVicherek.zip - Projekt diplomové práce v Tecnomatix Process Simulate v 15.1
- Program - Složka s vytvořeným programem
- BA_ready2_educate_cs_V13.pdf - Dokumentace KUKA ready to educate
- KUKA_KSS_8_5_Manual_for_integrators_en.pdf - Dokumentace KUKA KSS 8.5 manual for integrators